

Enhancing Object Detection in Self-Driving Cars Using a Hybrid Approach

Sajjad Ahmad Khan ¹ , Hyun Jun Lee ² and Huhnkuk Lim ^{1,*} 

¹ Computer Engineering Department, Hoseo University, Asan 31499, Republic of Korea; dr.sajjadkhan19@gmail.com

² AI Networks, Cheonan 31116, Republic of Korea; hjlee@ainetwork.co.kr

* Correspondence: slow63347@gmail.com

Abstract: Recent advancements in artificial intelligence (AI) have greatly improved the object detection capabilities of autonomous vehicles, especially using convolutional neural networks (CNNs). However, achieving high levels of accuracy and speed simultaneously in vehicular environments remains a challenge. Therefore, this paper proposes a hybrid approach that incorporates the features of two state-of-the-art object detection models: You Only Look Once (YOLO) and Faster Region CNN (Faster R-CNN). The proposed hybrid approach combines the detection and boundary box selection capabilities of YOLO with the region of interest (RoI) pooling from Faster R-CNN, resulting in improved segmentation and classification accuracy. Furthermore, we skip the Region Proposal Network (RPN) from the Faster R-CNN architecture to optimize processing time. The hybrid model is trained on a local dataset of 10,000 labeled traffic images collected during driving scenarios, further enhancing its accuracy. The results demonstrate that our proposed hybrid approach outperforms existing state-of-the-art models, providing both high accuracy and practical real-time object detection for autonomous vehicles. It is observed that the proposed hybrid model achieves a significant increase in accuracy, with improvements ranging from 5 to 7 percent compared to the standalone YOLO models. The findings of this research have practical implications for the integration of AI technologies in autonomous driving systems.

Keywords: artificial intelligence; autonomous vehicles; convolutional neural network; Faster R-CNN; YOLO



Citation: Khan, S.A.; Lee, H.J.; Lim, H. Enhancing Object Detection in Self-Driving Cars Using a Hybrid Approach. *Electronics* **2023**, *12*, 2768. <https://doi.org/10.3390/electronics12132768>

Academic Editors: Felipe Jiménez and Mahmut Reyhanoglu

Received: 7 May 2023

Revised: 16 June 2023

Accepted: 20 June 2023

Published: 21 June 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The automotive industry is undergoing a rapid transformation with the integration of artificial intelligence (AI) technologies [1,2]. The global market for AI in vehicles is projected to experience significant growth, with AI software for automobiles expected to reach USD 6.6 billion by 2025, expanding at an average annual rate of 36.15% [3,4].

Object detection plays a crucial role in autonomous driving systems, enabling self-driving cars to perceive and classify objects in their surroundings accurately. Two main strategies are commonly employed for object detection, namely, the two-stage and single-stage approaches. Two-stage approaches, such as the region-based convolutional neural network (RCNN) [5] and the Faster R-CNN [6], are widely adopted in both academic research and industry due to their high accuracy. These methods use a region proposal network (RPN) to generate potential object proposals, followed by a classification and bounding box regression stage. However, they often fall short of meeting the real-time processing requirements of object detection in vehicular environments [7].

On the other hand, single-stage approaches, exemplified by you only look once (YOLO) [8] and the single shot detector (SSD) [9], are renowned for their speed. These methods directly predict object classes and bounding box coordinates from a single evaluation of the network. They achieve real-time performance but may sacrifice some accuracy

in certain scenarios [10]. Despite their advantages, detecting objects of varying sizes and under challenging driving conditions remains a significant challenge for these single-stage approaches [11].

To overcome the limitations of both two-stage and single-stage methods, there is a growing interest in developing hybrid approaches that leverage the strengths of both paradigms. These hybrid models aim to achieve a balance between accuracy and speed in object detection tasks. By incorporating ideas from both YOLO and Faster R-CNN, it is possible to design a hybrid architecture that provides improved accuracy while maintaining real-time processing capabilities. The proposed hybrid approach combines the efficient object detection of YOLO for bounding box selection with the region of interest (RoI) pooling mechanism from Faster R-CNN for accurate segmentation and classification [12]. Skipping the RPN from the Faster R-CNN architecture further accelerates the processing time without significant loss in accuracy [13].

However, the need for an improved object detection AI model for autonomous vehicles arises due to fundamental challenges faced in object recognition, such as image diffusion, blurring, and motion, particularly in the context of a smart city environment. Moreover, practical limitations, including limited computer resources and latency issues, can hinder the ability to quickly and accurately identify objects in specific environments [14]. Therefore, several recent studies have proposed novel techniques to enhance the performance of hybrid object detection models for self-driving cars. For instance, [15] proposed a deep-learning-based hybrid framework that uses an optimized YOLOv4 model to detect 10 kinds of object and a fine-tuned part affinity fields approach to estimate the pose of pedestrians. In [16], a survey of deep learning applications for object detection and scene perception in autonomous vehicles is presented, and challenges and open issues are discussed. The authors of [17] introduced an adaptive fusion method that combines YOLOv3 with semantic segmentation networks to improve small object detection in complex scenes. Similarly, the authors of [18] developed a multi-scale feature fusion network that integrates YOLOv3 with Faster R-CNN to enhance large-scale object detection in urban traffic scenarios. Additionally, the authors of [19] proposed a lightweight hybrid model that uses YOLOv3-tiny as a backbone network and adds an RoI pooling layer to reduce computational complexity and memory consumption.

This paper presents a valuable contribution to the field of object detection for self-driving cars by proposing a hybrid approach that combines the strengths of the YOLO and Faster R-CNN frameworks to achieve higher accuracy and practical speed. The proposed method uses YOLO to detect all objects in a given frame and selects the boundary box around each object. The region of interest (RoI) pooling is then borrowed from Faster R-CNN and used for object segmentation and classification, leading to enhanced accuracy. The region proposal network (RPN) is skipped to reduce the processing time. The proposed model is trained on a large dataset of 10,000 labeled traffic images captured during local driving scenarios to enhance detection accuracy in vehicular environments further and increase its effectiveness in real-world situations.

By using a combination of the YOLO and Faster R-CNN frameworks, along with incorporating a local traffic dataset, the proposed hybrid approach represents a significant step towards achieving higher accuracy than YOLOv5 and YOLOv7, while supporting a practical real-time speed for autonomous vehicles. Furthermore, the proposed hybrid approach is considered to be an efficient object detection method in vehicular environments.

The rest of the paper is organized as follows: Section 2 reviews the related work on object detection for self-driving cars. Section 3 presents the proposed hybrid approach for object detection and its components. Section 4 presents the experimental results, including performance metrics and an evaluation of the proposed approach using real-world data. Finally, Section 5 provides a summary of the key findings and conclusions, along with suggestions for future research directions.

2. Related Work

Object detection and recognition are essential tasks in the field of autonomous driving and vehicular communication. In recent years, deep-learning-based object detection techniques have become popular due to their high accuracy and performance. In this section, we discuss some of the most recent and relevant papers related to the object detector in vehicular communication.

The authors of [20] proposed a real-time multi-task framework for pedestrian and vehicle detection which is based on YOLOv5. The proposed framework uses a single neural network to perform both tasks, which significantly reduces the computational complexity and improves the detection accuracy. The experiments were conducted on the KITTI dataset, and the results showed that the proposed framework outperforms state-of-the-art methods in terms of accuracy and speed. The main advantage of this framework is its ability to detect both pedestrians and vehicles in real-time, which is crucial for the safety of autonomous vehicles. However, the limitation of this framework is that it only focuses on pedestrian and vehicle detection and does not consider other objects that may be present on the road.

Another paper [21] proposed an efficient hybrid deep learning architecture for object detection and tracking in autonomous vehicles. The proposed architecture is a combination of a two-stage object detector and a tracking model. The two-stage object detector is based on Faster R-CNN, while the tracking model is based on the Kalman filter. The experiments were conducted on the KITTI dataset, and the results showed that the proposed architecture outperforms state-of-the-art methods in terms of accuracy and speed. The main advantage of this architecture is its ability to detect and track objects in real-time, which is crucial for the safety of autonomous vehicles. However, the limitation of this architecture is its complexity, which may require high computational resources.

In addition, another paper [22] proposed a novel hybrid CNN-LSTM framework for object detection in autonomous vehicles. The proposed framework is a combination of a convolutional neural network (CNN) and a long short-term memory (LSTM) network. CNN is used for feature extraction, while LSTM is used for sequence modeling. The experiments were conducted on the KITTI dataset, and the results showed that the proposed framework outperforms state-of-the-art methods in terms of accuracy and speed. The main advantage of this framework is its ability to model the temporal dependencies in object detection, which is crucial for the safety of autonomous vehicles. However, the limitation of this framework is its high computational complexity, which may require high computational resources.

Another paper [23] was written by A. Ojha et al., who proposed a hybrid intelligent model for real-time vehicle detection and tracking. The proposed model consists of a YOLOv3 object detector, a CNN-based tracker, and a Kalman filter (KF) estimator. The model was evaluated using the UA-DETRAC benchmark dataset. The results show that the proposed model achieves state-of-the-art performance in terms of detection accuracy and tracking efficiency. In this paper, a hybrid model is used to combine different techniques to achieve high accuracy and efficiency. Furthermore, the model is evaluated using a widely used benchmark dataset. However, the model is evaluated only on one dataset, which limits its generalizability to other datasets. Similarly, the model is designed specifically for vehicle detection and tracking and may not be applicable to other types of objects.

Jia, X. et al. present a novel object detection algorithm for autonomous driving based on improved YOLOv5 in [24]. The algorithm enhances the accuracy and speed of the model by using structural re-parameterization, neural architecture search, small object detection layer, and coordinate attention mechanism. The algorithm is tested on the KITTI dataset and shows superior performance compared to other methods in terms of accuracy and speed. However, the algorithm also has some drawbacks, such as requiring a large amount of training data and being sensitive to noise and occlusion.

The reviewed papers propose various approaches for object detection and tracking, including traditional methods, deep-learning-based methods, and hybrid methods. However,

these methods have some limitations, such as limited accuracy and efficiency, an inability to handle complex environments, and difficulty in generalizing to various types of objects. The comparisons are given in Table 1.

Table 1. Comparison of different object detection models used in vehicular communication.

Ref.	Year	Methodology	Advantages	Limitations
[20]	2021	Hybrid YOLOv3	High accuracy and real-time performance	It is limited to detecting certain types of objects.
[21]	2022	Hybrid RetinaNet	Improved accuracy and robustness	It requires a significant amount of computational resources.
[22]	2022	Hybrid CenterNet	Efficient computation and high accuracy	It can be used only for small-scale detection tasks.
[23]	2022	Hybrid SSD	Improved accuracy and efficiency	It has the capability to detect fewer types of objects.
[24]	2023	YOLOv5	High accuracy and fast computation	It requires a large amount of training data and is limited to blurred and unclear objects.
Proposed	2023	Hybrid YOLO and Faster R-CNN	High accuracy, and wide range of object detection capabilities	It requires more computational resources than YOLOv5 and YOLOv7.

To overcome these limitations, this paper proposes a hybrid object detection model for vehicular communication that utilizes the features of both the popular models YOLO and Faster R-CNN to improve accuracy in practical real-time object detection scenarios. In addition, the proposed hybrid model is designed to address the challenges of real-time object detection in vehicular environments. One of the key benefits of our model is its ability to detect and recognize a wide range of objects commonly found in such environments. Specifically, our model is trained to detect and classify eleven different types of object, including number plates, persons, cars, motorcycles, buses, big and small trucks, traffic lights, traffic signs, faces, and crosswalks. In contrast, other object detection models, such as YOLO and SSD, are limited in their ability to detect such a diverse range of objects.

3. Proposed Hybrid Model for Object Detection

This section presents a hybrid object detection model for vehicular communication that combines the YOLO and Faster R-CNN architectures to detect and track objects accurately and efficiently in real-world driving scenarios. The proposed model utilizes collected data and labeled datasets for autonomous vehicle training.

Figure 1 illustrates the object detection procedure of the proposed model used in autonomous vehicles. The process starts with collecting real-time videos recorded by the local driving cars. The real-time videos are then displayed on the driver's video viewer. The hybrid object detection model is used to extract features from the videos. The model detects different objects using their unique features. The region of interest (RoI) is then activated, and the area is detected with pixel values. Finally, the objects are classified according to their type using the extracted features.

The proposed hybrid model development process consists of several steps, as shown in Figure 2. The first step is data collection, where the data are collected from driving scenarios to train the object detection model. The collected data are then preprocessed to make them

suitable for training. This includes resizing images, normalizing pixel values, and dividing the dataset into training and validation sets to evaluate the model’s performance.

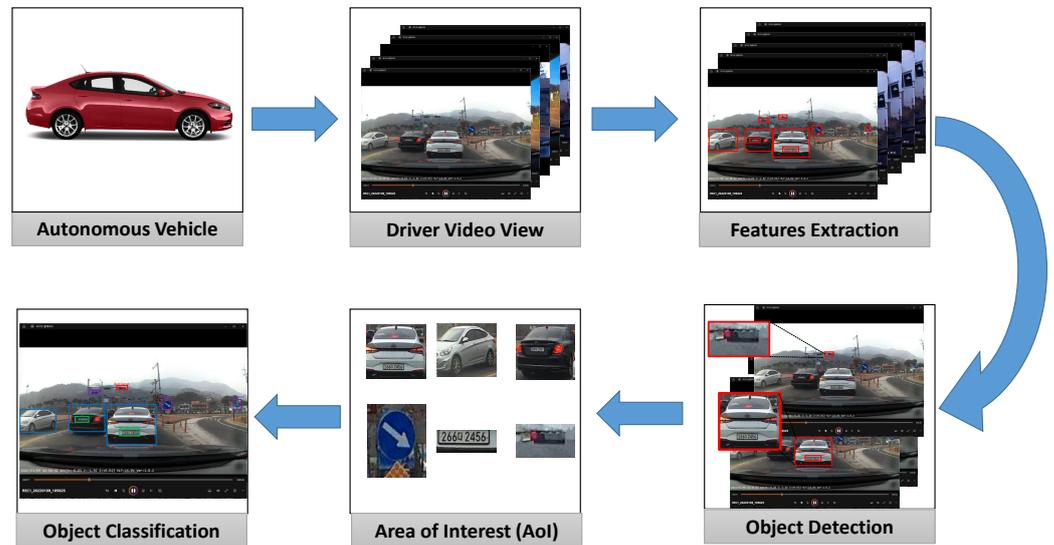


Figure 1. Object detection procedure of proposed hybrid model for autonomous vehicles.

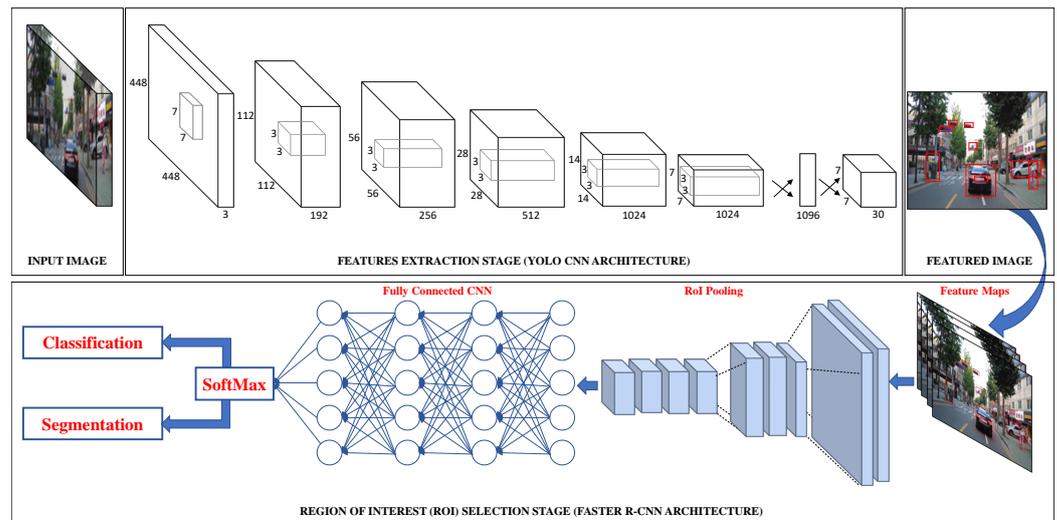


Figure 2. Network structure diagram of the proposed hybrid model.

The next step involves implementing the YOLO model to extract bounding boxes for objects from the preprocessed images. YOLO provides a fast and approximate object detection solution, making it suitable for real-time applications. Typically, each bounding box is represented by a set of six parameters: P_c , B_x , B_y , B_w , B_h , and C_n , where P_c is the probability of an object belonging to a particular class, and B_x , B_y , B_w , B_h , and C_n represent the center coordinates, width, height, and class number, respectively. Additionally, each bounding box is assigned a confidence prediction, which is computed as the intersection over union (IoU) between the predicted box and the ground truth box. The IoU can be obtained using the following equation:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{1}$$

After obtaining the bounding boxes and their confidence predictions, the featured images are passed through the Faster R-CNN block. In this block, the region of interest (RoI) pooling is applied to extract the region's features. The RoI can be expressed as:

$$\text{RoI pooling}(P_{i,k}) = \frac{1}{n} \sum_{j=1}^n P_{i,k}(\text{pool}(h_j, w_j, x_j, y_j)) \quad (2)$$

where $P_{i,k}$ is the output feature map of the last convolutional layer for the i -th RoI, and k is the index of its channels. n is the number of sub-windows to be divided in the RoI, $P_{i,k}(\text{pool}(h_j, w_j, x_j, y_j))$ is the activation value of the k -th channel at the j -th sub-window in the RoI, and $\text{pool}(h_j, w_j, x_j, y_j)$ is the sub-window of the RoI with height h_j , width w_j , and coordinates (x_j, y_j) .

The RoI-pooled features are then fed into a fully connected CNN, which performs classification and segmentation tasks. The activation function used in the CNN layers is the rectified linear unit (ReLU), which is defined as:

$$\text{ReLU}(x) = \max(0, x) \quad (3)$$

where x is the input to the activation function, this function returns x if x is positive and returns 0 otherwise.

The softmax loss factor is utilized in the fully connected layer to calculate the loss and optimize the model during the training process. This loss function measures the discrepancy between the predicted class probabilities and the ground truth labels. It encourages the model to assign higher probabilities to the correct object class and lower probabilities to incorrect classes.

The softmax cross-entropy loss factor equation can be expressed as:

$$\text{Loss}_{cls} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C y_{ij} \log(p_{ij}) \quad (4)$$

where N is the number of samples in the training batch, C is the total number of object classes, y_{ij} is the ground truth label indicating whether sample i belongs to class j , and p_{ij} is the predicted probability of sample i belonging to class j .

Finally, the model achieves accurate classification and segmentation results, enabling robust object detection in vehicular environments.

It is worth mentioning that the RPN is skipped during the Faster R-CNN step. Since the RPN is only used for generating region proposals and the YOLO framework has already detected all the objects in the frame, the RPN can be skipped to reduce the processing time, resulting in high speed. This is because the RPN involves a considerable amount of computation and can be time consuming, especially when dealing with large datasets. Therefore, by using only the RoI pooling and eliminating the RPN, the proposed hybrid approach can achieve a practical real-time speed without compromising the accuracy of the detection.

By minimizing this loss factor during the training process, our hybrid model learns to optimize the softmax classifier, improving its ability to classify objects accurately in real-world driving scenarios.

By combining the YOLO and Faster R-CNN architectures, the proposed model achieves high accuracy and efficiency in real-time object detection and classification, particularly in vehicular communication. YOLO is utilized for object detection and bounding box generation, and the RoI pooling is borrowed from Faster R-CNN for segmentation and classification. This hybrid results in a model capable of accurately and efficiently detecting and classifying objects in real-time scenarios. The proposed hybrid model is given in Figure 3 that combines the features of the YOLO and Faster R-CNN architectures. The model consists of several key components that work together to achieve accurate and real-time object detection in autonomous driving scenarios.

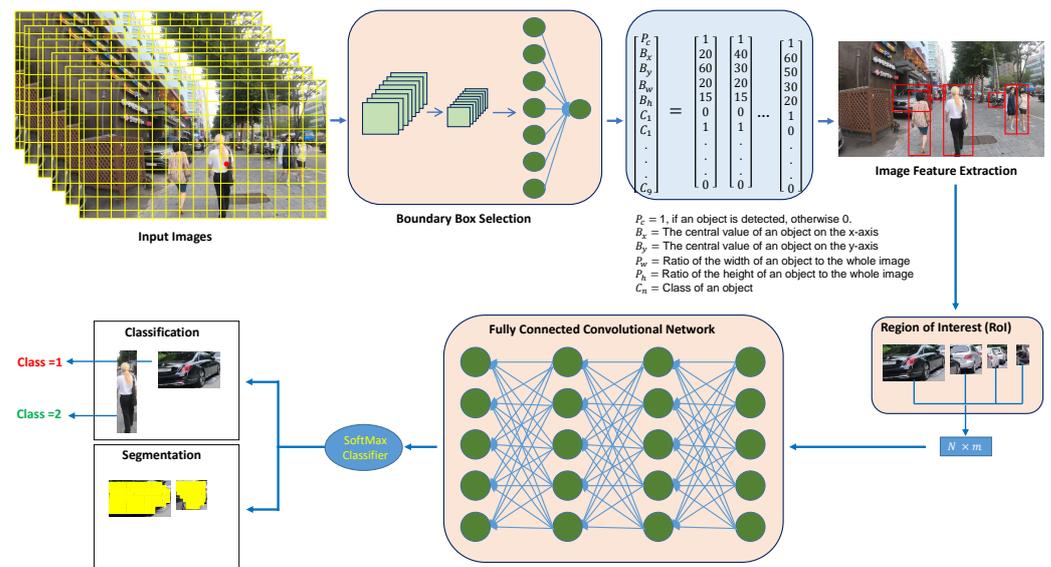


Figure 3. The proposed hybrid object detection model for real-time object detection in autonomous driving scenarios.

To ensure the accuracy and effectiveness of our model, we trained it on a large dataset of images containing eleven different types of objects, such as, number plates, persons, cars, motorcycles, buses, big and small trucks, traffic lights, traffic signs, faces, and crosswalks. This dataset is carefully created to include a wide range of scenarios and lighting conditions that are commonly encountered in vehicular environments. By training the proposed model on such a diverse dataset, it is able to accurately detect and classify these objects in real-time scenarios. Overall, the ability of our proposed hybrid model to detect and recognize a wide range of objects commonly found in vehicular environments makes it better than the other object detection models.

4. Experimental Results

This section presents the evaluation and performance of different object detection models for detecting road traffic objects for self-driving cars. A dataset of 10,000 images was used to train and test all the object detection models. The dataset included various types of vehicles, traffic signals, road signs, pedestrians, and number plates. A sample of our dataset is given in Figure 4.

The images were labeled using the DarkLabel tool, and both the training and testing phases were conducted on the NVIDIA Jetson Xavier NX platform, which is a promising option for use in self-driving cars. To evaluate the performance of each model, we measured the inference time and mean average precision (mAP), as well as the confidence and recall values. Additionally, we calculated the average image processing speed to compare the speed of our proposed hybrid model with other state-of-the-art models.

The performance of four distinct object detection models, namely, the proposed hybrid model, Faster R-CNN, YOLOv5, and YOLOv7, is depicted in Figure 5 in terms of mAP and inference time (ms). The proposed hybrid model achieves the highest mAP values compared to all other models when the inference time is between 4 ms and 44 ms. Specifically, the proposed hybrid model achieved an mAP of approximately 26% at an inference time of 4 ms, an mAP of 66% at 24 ms, and an mAP of 73% at 44 ms. Conversely, the Faster R-CNN model attained an mAP of approximately 45% at an inference time of 12 ms, while at an inference time of 57 ms it achieved an mAP of around 76%. The YOLOv5 and YOLOv7 models achieved mAP of around 19% and 20%, respectively, at an inference time of 2 ms. On the other hand, at an inference time of 33 ms, these models achieved mAP of approximately 67.5% and 70%, respectively.

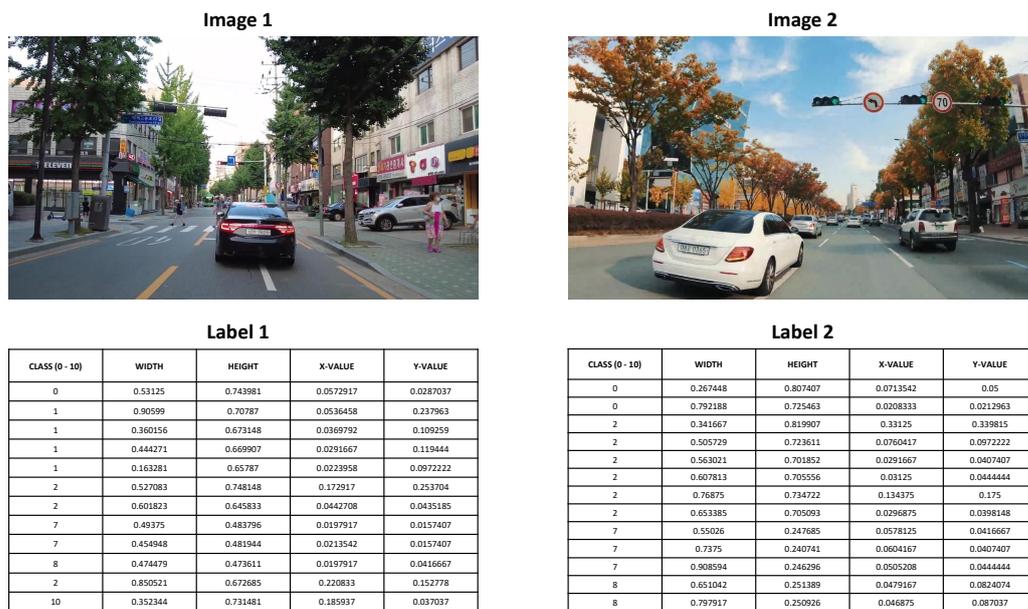


Figure 4. A sample of our dataset, including images and labels.

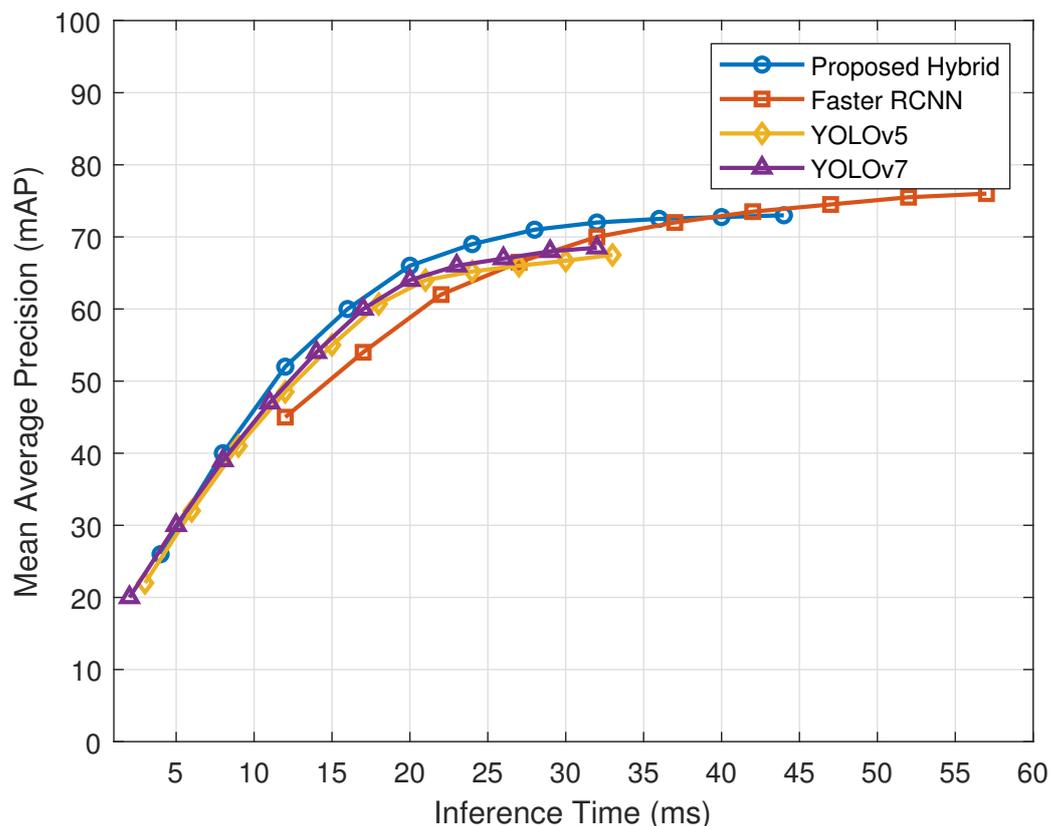


Figure 5. Comparison of mean average precision (mAP) and inference times for state-of-the-art object detection models.

Figure 6 represents a comparison of object detection models with respect to their performance in terms of confidence and recall. Confidence is the probability that detection is correct, while recall is the fraction of true positive detections out of all possible positive detections. The x-axis represents the recall values in percentages ranging from 30% to 100%, and the y-axis represents the confidence values in percentages ranging from 0% to 100%. Each curve in the figure represents the variation in confidence with respect to recall for each

model. Figure 6 shows that the proposed hybrid model achieves the highest confidence values compared to all other models for the same recall values. The YOLOv7 model shows better performance than YOLOv5, but it has lower accuracy than the proposed hybrid model. YOLOv5 and YOLOv7 consume less time compared to Faster R-CNN and the proposed hybrid model, but they have lower accuracy. On the other hand, the Faster R-CNN achieves better accuracy than YOLOv7 and YOLOv5, but with the cost of processing speed. It is observed that the confidence of our proposed model is lower than that of Faster R-CNN when the recall exceeds 65%. We believe this discrepancy can be attributed to the trade-off between recall and confidence in our model.

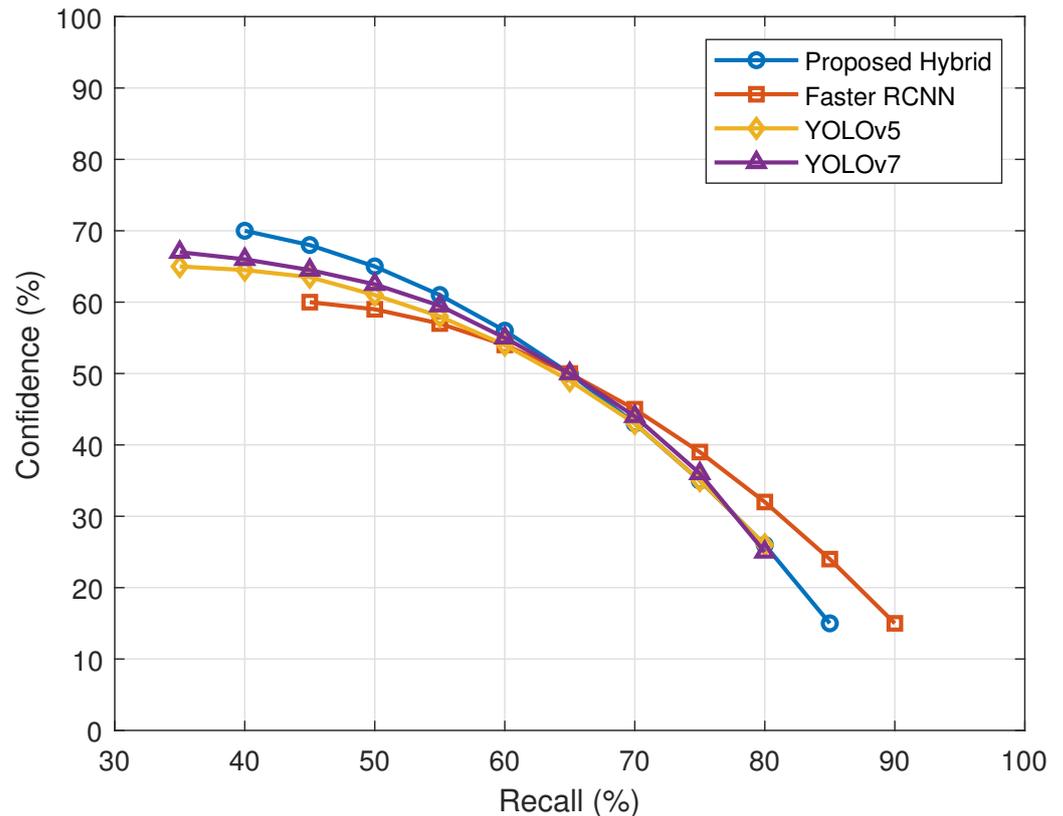


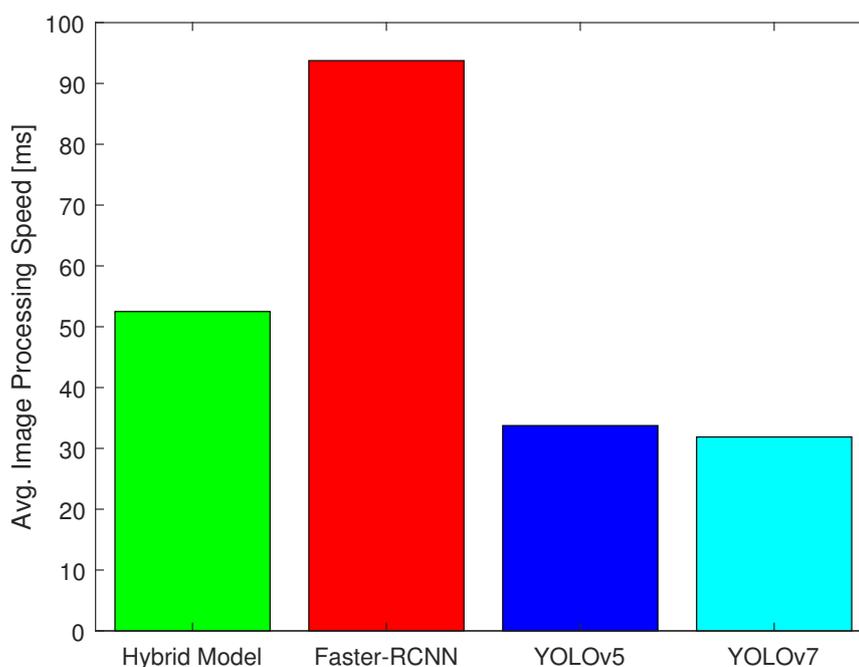
Figure 6. Comparison of state-of-the-art object detection models with respect to confidence and recall values.

Figure 7 illustrates a comparison of the processing speed of four different object detection models, with the average processing times for one image in milliseconds (ms) for each model. The proposed model achieves an average processing time of 52 ms per image, which ranks it as the third-fastest model among the four models compared. Among the four models, the Faster R-CNN model is the slowest, with an average processing time of 92 ms per image. Typically, for effective object detection in vehicular environments, processing times of at least 62.5 ms per image, equivalent to 16 frames per second, are required [25]. The YOLOv5 model ranks second in terms of speed, with an average processing time of 32 ms per image, while the YOLOv7 model is the fastest, with an average processing time of 30 ms per image. Table 2 provides a comparison of the object detection performance between the hybrid model, YOLOv5, YOLOv7, and Faster R-CNN.

Overall, the results demonstrate that the proposed hybrid model outperforms the other models in terms of accuracy, supporting practical real-time object detection applications in autonomous vehicles.

Table 2. Performance comparison between state-of-the-art object detection models.

Model	Accuracy (%)	Processing Time (ms)	Recall (%)
Hybrid Model	74.3	52	86
YOLOv5	67.5	32	80
YOLOv7	68.2	30	82
Faster R-CNN	79.1	92	90

**Figure 7.** Comparison of the processing speeds of state-of-the-art object detection models.

5. Conclusions

In this paper, we proposed a hybrid approach that combines the strengths of the YOLO and Faster R-CNN architectures to achieve both high accuracy and real-time speed for object detection in self-driving cars. By leveraging the YOLO framework for object detection and bounding box generation, and incorporating RoI pooling from Faster R-CNN for segmentation and classification, our hybrid approach offers improved accuracy compared to state-of-the-art models such as YOLOv5 and YOLOv7, with an enhancement of five to seven percent. We trained and evaluated our approach on a dataset of 10,000 labeled images from local driving scenarios, demonstrating its effectiveness in real-world conditions. The integration of advanced AI techniques, such as convolutional neural networks, in autonomous vehicles holds great promise for improving transportation systems' safety and efficiency. Future research can focus on utilizing larger and more diverse datasets to further evaluate the effectiveness of our approach across different autonomous vehicle platforms. Additionally, optimizing the implementation and exploring synergies with other AI-based technologies can further enhance the performance and safety of self-driving cars.

Author Contributions: All authors have contributed equally. All authors have read and agreed to the published version of the manuscript.

Funding: This work was sponsored by the Korean Government (Ministry of Science and ICT) through the National Research Foundation of Korea (NRF) under grant number 2021R1A2C1010481.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This research work was carried out at the Department of Computer Engineering, Hoseo University, Asan-si, South Korea. The authors would like to thank the University administration for providing the facilities during this research work.

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

References

1. Mankodiya, H.; Jadav, D.; Gupta, R.; Tanwar, S.; Hong, W.C.; Sharma, R. Od-xai: Explainable ai-based semantic object detection for autonomous vehicles. *Appl. Sci.* **2022**, *12*, 5310. [[CrossRef](#)]
2. Khan, S.A.; Lim, H. Novel Fuzzy Logic Scheme for Push-Based Critical Data Broadcast Mitigation in VNDN. *Sensors* **2022**, *22*, 8078. [[CrossRef](#)] [[PubMed](#)]
3. Bhavana, N.; Kodabagi, M.M. Exploring the Current State of Road Lane Detection: A Comprehensive Survey. *Int. J. Hum. Comput. Intell.* **2023**, *2*, 40–46.
4. Khan, S.A.; Lim, H. Push-Based Forwarding Scheme Using Fuzzy Logic to Mitigate the Broadcasting Storm Effect in VNDN. In Proceedings of the Artificial Intelligence and Mobile Services–AIMS 2022: 11th International Conference, Held as Part of the Services Conference Federation, SCF 2022, Honolulu, HI, USA, 10–14 December 2022; Proceedings; Springer: Berlin/Heidelberg, Germany, 2022; pp. 3–17.
5. Balasubramanian, R. Region-Based Convolutional Neural Network (RCNN). Available online: <https://medium.com/analytics-vidhya/region-based-convolutional-neural-network-rcnn-b68ada0db871> (accessed on 29 March 2023).
6. Rani, S.; Ghai, D.; Kumar, S. Object detection and recognition using contour based edge detection and fast R-CNN. *Multimed. Tools Appl.* **2022**, *81*, 42183–42207. [[CrossRef](#)]
7. Güney, E.; Bayilmiş, C. An implementation of traffic signs and road objects detection using faster R-CNN. *Sak. Univ. J. Comput. Inf. Sci.* **2022**, *5*, 216–224. [[CrossRef](#)]
8. Jiang, P.; Ergu, D.; Liu, F.; Cai, Y.; Ma, B. A Review of Yolo algorithm developments. *Procedia Comput. Sci.* **2022**, *199*, 1066–1073. [[CrossRef](#)]
9. Cheng, L.; Ji, Y.; Li, C.; Liu, X.; Fang, G. Improved SSD network for fast concealed object detection and recognition in passive terahertz security images. *Sci. Rep.* **2022**, *12*, 1–16. [[CrossRef](#)] [[PubMed](#)]
10. Sindhwani, N.; Verma, S.; Bajaj, T.; Anand, R. Comparative analysis of intelligent driving and safety assistance systems using YOLO and SSD model of deep learning. *Int. J. Inf. Syst. Model. Des.* **2021**, *12*, 131–146. [[CrossRef](#)]
11. Abdullah, S.T.; AL-Nuaimi, B.T.; Abed, H.N. A survey of deep learning-based object detection: Application and open issues. *Int. J. Nonlinear Anal. Appl.* **2022**, *13*, 1495–1504.
12. Pazhani, A.A.J.; Vasanthanayaki, C. Object detection in satellite images by faster R-CNN incorporated with enhanced ROI pooling (FrRNet-ERol) framework. *Earth Sci. Inform.* **2022**, *15*, 553–561. [[CrossRef](#)]
13. Xiao, Y.; Wang, X.; Zhang, P.; Meng, F.; Shao, F. Object detection based on faster R-CNN algorithm with skip pooling and fusion of contextual information. *Sensors* **2020**, *20*, 5490. [[CrossRef](#)] [[PubMed](#)]
14. Wei, Z.; Zhang, F.; Chang, S.; Liu, Y.; Wu, H.; Feng, Z. Mmwave radar and vision fusion for object detection in autonomous driving: A review. *Sensors* **2022**, *22*, 2542. [[CrossRef](#)] [[PubMed](#)]
15. Li, Y.; Wang, H.; Dang, L.M.; Nguyen, T.N.; Han, D.; Lee, A.; Jang, I.; Moon, H. A deep learning-based hybrid framework for object detection and recognition in autonomous driving. *IEEE Access* **2020**, *8*, 194228–194239. [[CrossRef](#)]
16. Gupta, A.; Anpalagan, A.; Guan, L.; Khwaja, A.S. Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. *Array* **2021**, *10*, 100057. [[CrossRef](#)]
17. Li, J.; Liang, X.; Shen, S.; Xu, T.; Feng, J.; Yan, S. Scale-aware fast R-CNN for pedestrian detection. *IEEE Trans. Multimed.* **2017**, *20*, 985–996. [[CrossRef](#)]
18. Zeng, N.; Wu, P.; Wang, Z.; Li, H.; Liu, W.; Liu, X. A small-sized object detection oriented multi-scale feature fusion approach with application to defect detection. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 3507014. [[CrossRef](#)]
19. Zhou, L.; Deng, G.; Li, W.; Mi, J.; Lei, B. A Lightweight SE-YOLOv3 Network for Multi-Scale Object Detection in Remote Sensing Imagery. *Int. J. Pattern Recognit. Artif. Intell.* **2021**, *35*, 2150037. [[CrossRef](#)]
20. Razali, H.; Mordan, T.; Alahi, A. Pedestrian intention prediction: A convolutional bottom-up multi-task approach. *Transp. Res. Part C Emerg. Technol.* **2021**, *130*, 103259. [[CrossRef](#)]
21. Choudhury, S.; Karthik Pai, B.; Hemant Kumar Reddy, K.; Roy, D.S. A Hybrid CNN Real-Time Object Identification and Classification Approach for Autonomous Vehicles. In *Intelligent Systems: Proceedings of ICMIB 2021*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 485–497.
22. Kortli, Y.; Gabsi, S.; Voon, L.F.L.Y.; Jridi, M.; Merzougui, M.; Atri, M. Deep embedded hybrid CNN–LSTM network for lane detection on NVIDIA Jetson Xavier NX. *Knowl.-Based Syst.* **2022**, *240*, 107941. [[CrossRef](#)]
23. Ojha, A.; Sahu, S.P.; Dewangan, D.K. VNet: Vehicle detection network using computer vision and deep learning mechanism for intelligent vehicle system. In Proceedings of the Emerging Trends and Technologies on Intelligent Systems: ETTIS 2021, Noida, India, 4–5 March 2021; Springer: Berlin/Heidelberg, Germany, 2022; pp. 101–113.

24. Jia, X.; Tong, Y.; Qiao, H.; Li, M.; Tong, J.; Liang, B. Fast and accurate object detector for autonomous driving based on improved YOLOv5. *Sci. Rep.* **2023**, *13*, 9711. [[CrossRef](#)] [[PubMed](#)]
25. Mehta, R.; Ozturk, C. Object detection at 200 frames per second. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.