Performance Comparison of Object Detection Models for Road Sign Detection Under Different Conditions

Zainab Fatima[1], M. Hassan Tanveer[2], Hira Mariam[3], Razvan Cristian Voicu[4], Tanazzah Rehman[5], Rizwan Riaz[6]
Department of Software Engineering, NED University of Engineering & Technology, Karachi 75270, Pakistan[1,5,6]
Department of Robotics and Mechatronics Engineering, Kennesaw State University, Marietta, GA 30060, USA[2,4]
Department of Telecommunications Engineering, NED University of Engineering & Technology, Karachi 75270, Pakistan[3]

***Abstract*—During driving, drivers often overlook the traffic signs along the roads compromising road safety and increasing the risk of accidents. To address this, artificial intelligence (AI) and deep learning techniques are employed, taking into consideration the improvement of advances in Artificial Neural Networks (ANNs) and image processing for robust road sign detection. In this work, we compare the performance of existing state-of-the-art object detection models for road sign detection, including YOLOv8, YOLOv9, RTMDet, Faster-RCNN and RetinaNet, using a large dataset of images of road signs. These models are fine-tuned and hyperparameters are optimized with varied settings like auto-orientation and augmentation during the preprocessing and training phase. The models are then tested, and key performance indicators such as mean average precision (mAP), number of inferences performed per second [frames per second (fps)], and total loss are evaluated. Our study reaffirms the earlier findings in which YOLOv9 and YOLOv8 outperform other detectors in real-time detection tasks because they are faster in inference or prediction than most detectors, but with a compromise in accuracy, as highlighted by the fast fps rates. In contrast, RTMDet is fast and reliable, making it a highly effective option for detecting various road signs. The insights presented in this research are useful in identifying the appropriateness and drawbacks of each model, thereby benefiting from the selection of the best suited model for real-world applications, such as autonomous vehicles or self-driving cars.**

Keywords*—*Artificial intelligence; artificial neural networks; image processing; deep learning; road signs detection

I. Introduction

A. Background

Over the last few years, the use of computer vision and object detection has become prominently efficient because of the shift in deep learning models. Such advancements have embellished object identification in both pictures and videos with great precision and efficacy. The relevance of an object being detectable by a vehicle increases with the overall progress in autonomous vehicles and smart and connected roads. Road sign detection has the necessary accuracy and operational capacity for serving its purpose in ensuring the safety for autonomous vehicles and advanced driver assistance systems ADAS.

B. Current State of Object Detection Models

This research paper presents a thorough comparative analysis of five leading object detection models: YOLOv9, YOLOv8, RetinaNet, RTMDet, and Faster R-CNN [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19]. Each of these models epitomizes state-of-the-art techniques in object detection, boasting unique strengths and characteristics that influence their performance in identifying objects within visual data.

- YOLO: You Only Look Once (YOLO) series, specifically YOLOv9 and YOLOv8, are most famous with real-time object detection, and that is why such models are applicable in areas that require high response rates [20], [21], [22]. They are particularly remarkable in terms of speed and efficiency of detection, which is crucial in contexts, such as autonomous driving, where minimizing detection time is of utmost importance.
- RetinaNet: Distinguished by its ability to accurately detect objects of varying sizes within images, making it a robust choice for scenarios involving diverse object scales. RetinaNet employs a Feature Pyramid Network (FPN) and a focal loss function to address the class imbalance and enhance detection accuracy [23], [24].
- Faster R-CNN: Widely recognized for its precise localization of objects, achieved through a region-based convolutional neural network that meticulously analyzes different regions within an image to ensure accurate detection [25], [26].
- RTMDet: RTMDet is known for its robustness in managing complex scenes and occlusions, demonstrating a superior ability to detect objects even under challenging conditions [27]. RTMDet integrates a modified ResNet-50 backbone with spatial and feature alignment modules to optimize detection performance.

C. Motivation

While YOLO series has been the object detection models that has made remarkable progress, there is a lack of extensive comparative analysis that is not only centered on the recent versions of the YOLO models (YOLOv9 and YOLOv8) but other cutting-edge models like RetinaNet, RTMDet, and Faster RCNN. Earlier works have mostly focused on giving attention to overall object identification or testing only a few of these models under various circumstances, which really did not enable researchers to make a Comparative analysis of these models for the purpose of road sign identification. This research intends to address this gap by comparing these models in detail and side by side in the context of road sign detection focusing on several measures and metrics and with an eye to useful recommendations for implementation in self-driving car technologies and as well in the driver assistant systems.

D. Contributions

The purpose of this work is to advance the prior art related to object detection and present a comprehensive and comparative evaluation of five of the most relevant approaches. When assessing the accuracy of each model, we have apply a set of unified metrics stemmed from a dataset of road signs, which provides useful information on the merits and drawbacks of each model. It can help researchers and practitioners on how best to choose the most suitable object detection model for intelligent transportation systems and autonomous driving.

This paper presents a comprehensive comparison of five prominent object detection architectures: YOLOv8, YOLOv9, RTMDet, RetinaNet, and Faster RCNN for the road sign detection study without leaving out the factors such as mAP and inference time. This paper is organised as follow: Section II presents the literature review relevant to this work, where we discuss the current state of object detection algorithms as well as road sign detection. Section III describes the three step model architectures of the aforementioned algorithms while Section III(B) explains the design principles and features of these algorithms. Section IV explains the experimental setup, procedures on how the datasets were prepared, training configurations that were used for training the models as well as the evaluation metrics that were employed in the assessment of the performance of each model. Results and discussion are discussed in Section V. Lastly, Section VI of this paper offers the conclusion and a discussion of the experiments to derive the factors that should be preferred while choosing between accuracy, speed, and confidence in detecting a road sign from the ones listed above or some other algorithms as per the parameters that will be followed strictly at the time of implementing the vision based systems.

II. Literature Review

The recognition of road signs is one of the critical segments of Intelligent Transportation Systems (ITS) performance, so its function is significant for providing road safety and effectiveness. In addition, as the deep learning starts to become more popular, object detection models show great potential in recognizing road sign. Road sign detection is revealed in this literature review as a line of research that experienced significant development in the last five years. In this section, the object detection models are introduced and analyzed based on the approach used, efficiency, effectiveness, and the kind of application which is road sign detection. The given review culminates the prior studies in order to perceive common aspects, issues, and opportunities in order to further establish more reliable and accurate type of road signs detection systems.

- This research enhanced YOLOv8 by adding blur and noise, and incorporating an asymptotic feature pyramid network, which improved the detection of small target objects. It achieved a 3.31% increase in mean Average Precision (mAP) and a 3.59% increase in recall on the TT100K dataset. These improvements were confirmed through ablation studies, highlighting the contributions of both data augmentation and the AFPN enhancements [28].
- Enhanced YOLOv8 algorithm for traffic sign recognition using the Kaggle dataset incorporated Cross-Stage Partial connection and Path Aggregation Network, achieving 80% accuracy, 64% precision, and 65.67% recall on test data. The use of stochastic gradient descent optimization and dropout helped curb overfitting, demonstrating the model's efficacy in complex traffic sign analysis [29].
- In a comparative study, YOLOv5 demonstrated superior performance over SSD in traffic sign recognition, achieving a mAP@0.5 of 97% and processing images at 30 FPS on the VOC dataset. SSD showed 90% accuracy but was significantly slower, processing images at only 3 FPS. YOLOv5's faster recognition speed and higher recall score make it a better solution for traffic sign recognition in Intelligent Transportation Systems [30].
- This research improved traffic sign detection using Faster R-CNN with enhancements like feature pyramid, deformable convolutions, and ROI alignment. Tested on the TT100K dataset, it achieved high accuracy rates of 92.6% in sunny conditions, 90.6% at sunset, and 86.9% in rainy conditions, outperforming SSD, YOLOv2, and even YOLOv5 in less favorable lighting and weather conditions [31].
- A real-time traffic sign detection system using Faster R-CNN was trained on a dataset of 1880 images from Turkey and the German Traffic Sign Recognition Benchmark (GTSRB). The model, trained over 10,000 iterations, achieved an accuracy of 88.99% and a total loss rate of 0.220, demonstrating robust detection capabilities [32].
- Zhu et al. developed a RetinaNet-based algorithm achieving a final F1 score of 0.923. While the model performed well in favorable conditions, it faced limitations in adverse weather, indicating a need for future research to improve performance under varied conditions [33].
- Inspired by YOLOv4 and YOLOv5, the TSR-SA method enhanced traffic sign detection by incorporating high-level features, a receptive field block-cross in the neck, and a Random Erasing-Attention data augmentation method. This approach achieved a state-of-the-art mAP of 90.2% on the TT100K dataset, surpassing YOLOv4 and YOLOv5-x, though it faced challenges with category imbalance [34].
- Senthilnayaki's system used Faster R-CNN for detection and Inception V2 for classification, improving detection in varied conditions by increasing anchor thickness and enhancing feature map resolution. This method proved effective and resilient, with plans for future enhancements to refine feature maps for more robust proposals [35].

- A system using the RetinaNet model was developed for real-time traffic light detection, employing transfer learning and modifications to anchor box sizes to detect small traffic lights. The model achieved a weighted mean average precision (mAP) of 0.54 with an execution time of 108ms, showing significant improvements in detection accuracy and speed [36].
- Jiang's work on an improved YOLOv5 network introduced a balanced feature pyramid and a global context block, enhancing feature fusion and extraction. Tested on the TT100K dataset, the model showed significant performance improvements with a 1.9% increase in mAP@.5, a 2.1% increase in mAP@.5:0.95, and improvements in precision and recall by 2.4% and 3.3%, respectively, proving its superiority for traffic sign detection [37].

III. Models Architecture

A. YOLOv8

The YOLOv8 model architecture represents a significant advancement in object detection, offering improved accuracy and speed over its predecessors. As shown in Fig. 1, the backbone of YOLOv8 is a modified CSPDarknet53 with 53 convolutional layers and cross-stage partial connections to enhance information flow [38].

The head of YOLOv8, includes multiple convolutional and fully connected layers responsible for predicting bounding boxes, objectiveness scores, and class probabilities. A notable feature of the head is the self-attention mechanism, which enables the model to focus on different parts of an image, adjusting the importance of various features.

YOLOv8 adopts an anchor-free detection approach, directly predicting object centers instead of offsets from predefined anchor boxes, thus speeding up post-processing steps like Non-Maximum Suppression (NMS). Additionally, YOLOv8 introduces changes in convolutions, such as replacing a 6x6 conv with a 3x3 conv in the stem and modifying the main building block, which enhances performance and efficiency.

B. YOLOv9

The YOLOv9 model architecture represents a significant advancement in real-time object detection, offering superior accuracy and efficiency compared to previous models. The backbone architecture utilizes Cross-stage Partial (CSP) Connection blocks to enhance gradient flow and reduce data loss during the feed-forward process, optimizing performance and accuracy. The head of YOLOv9 incorporates Programmable Gradient Information (PGI) and the Generalized Efficient Layer Aggregation Network (GELAN), which are crucial for preventing data loss, ensuring accurate gradient updates, and optimizing lightweight models for efficient object detection tasks (see Fig. 2 and 3). YOLOv9 also brings two new architectures, namely, YOLOv9-C and YOLOv9-E that improve accuracy and object detection efficiency in different applications through information choking and gradient flow rectification [39]. In addition, the YOLOv9 model has top-level accuracy and efficiency among the current models, including RT-DETR, YOLO-MS, or others due to the efficient use of conventional convolutions.

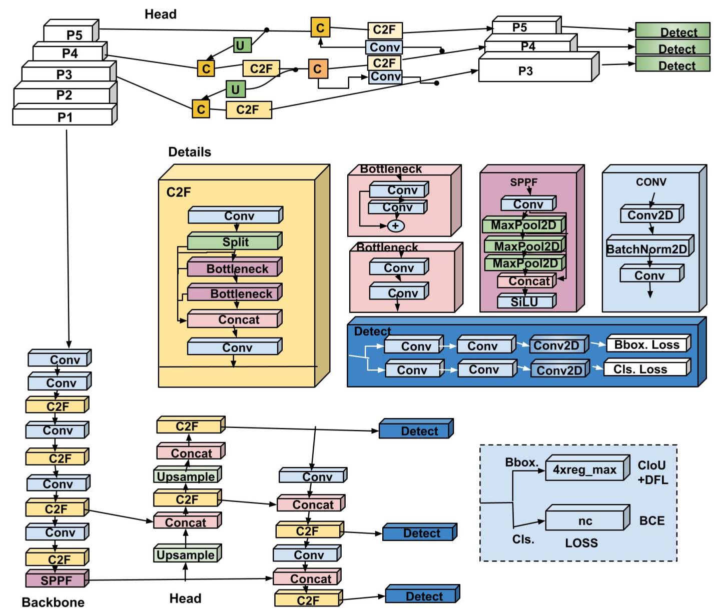

Fig. 1. YOLOv8 architecture.

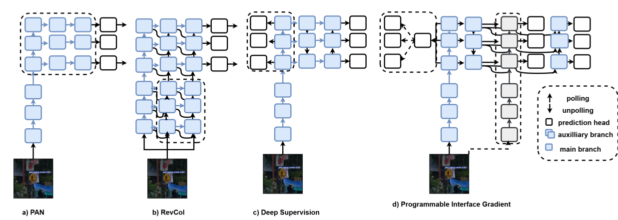

Fig. 2. YOLOv9 programmable gradient information.

C. RetinaNet

RetinaNet is a pioneering one-stage object detection model known for its exceptional performance in detecting objects at various scales. The unified network includes a backbone Convolutional Neural Network (CNN) and two task-specific subnets: the Classification Subnet and the Box Regression Subnet.

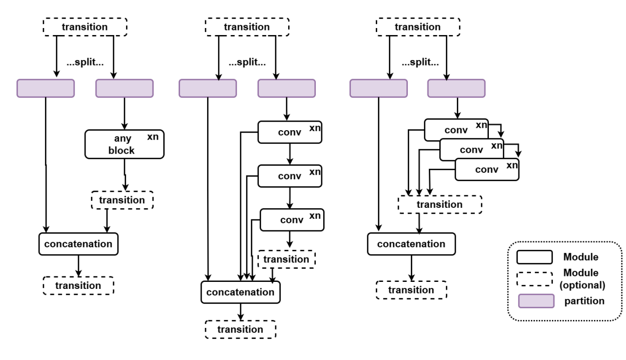

Fig. 3. YOLOv9 GELAN.

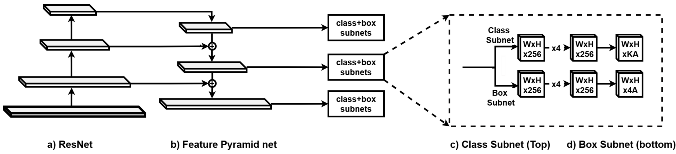

Fig. 4. RetinaNet structure.

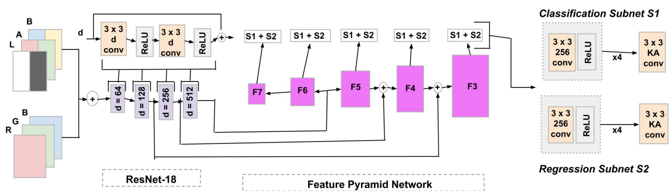

Fig. 5. Detailed structure.

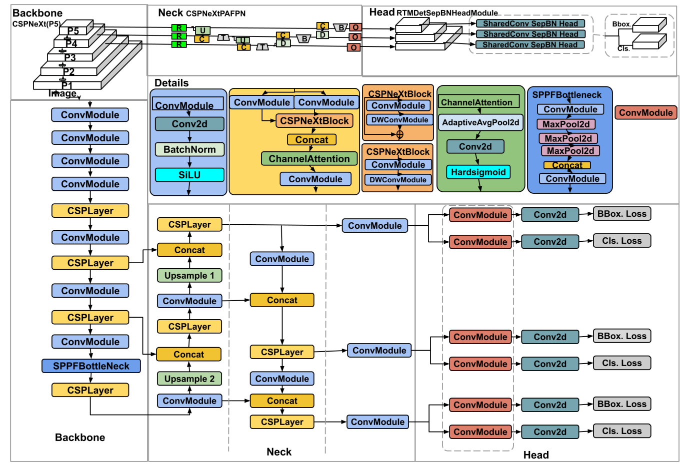

Fig. 6. RTMDet Architecture.

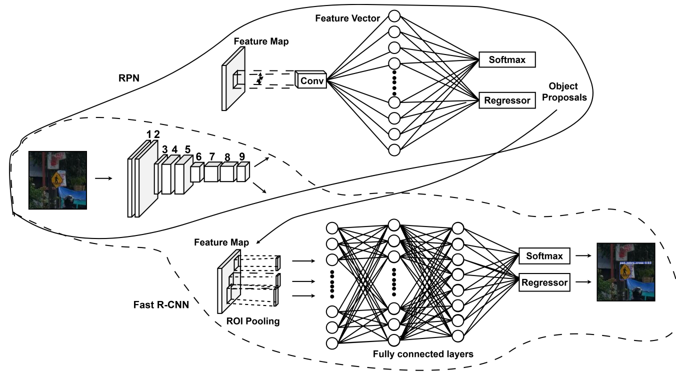

Fig. 7. Faster RCNN Architecture.

The backbone incorporates a Feature Pyramid Network (FPN), which generates a multi-scale feature pyramid by combining low-resolution semantically strong features with high-resolution features for accurate object detection. The Classification Subnet predicts object presence probabilities, while the Regression Subnet handles bounding box regression from anchor boxes, both using feature maps from the FPN (Fig. 4) and (Fig. 5). A key innovation is the Focal Loss function, which addresses class imbalance by assigning higher weights to hard examples, enhancing detection accuracy [24], [40]. Translation-invariant anchor boxes at different pyramid levels (P3 to P7) cover various scales and aspect ratios, enabling precise object localization and classification. During training, Stochastic Gradient Descent (SGD) is used with learning rate adjustments and data augmentation techniques like horizontal flipping to improve generalization.

D. RTMDet

RTMDet is an architecture will be proposed for real time object detection on the basis of the YOLO series of algorithms. It is used in scenarios where the detection of objects within images or videos in real-time is required, which makes this option highly effective in real-life designs. It uses ResNet-50 as its backbone after reducing the number of layers and parameters it uses for feature extraction. That is, some important modules: spatial attention module (SAM) is used to improve the feature extraction and the feature alignment module (FAM) to align the features from different scales [27].

The detection head of RTMDet uses a single convolutional layer to predict object bounding boxes and class probabilities, handling objects at three different scales: Depending on the amount of work, there are small, medium, and large offices. The RTMDet has three losses: CIoU loss that measures the loss of shapes and sizes, objectiveness loss that boosts completeness, and classification loss to maximize accuracy. For training, this model is trained on a large number of images and their corresponding labels and for the purpose of data augmentation has a considerable number of techniques incorporated. In inference, RTMDet replied it supports real-time object detection on still image and video, and it is deployable on different hardware environment such as GPU and TPU.

In this work, RTMDet presents itself with several advantages compared to the existing architectures of object detection such as high accuracy, high speed along with lesser computational complexity. Due to these characteristics, the technology is applied in self-driving cars, the monitoring systems, and robots among other areas. The above mentioned architecture including details of its subcomponents can be evident from Fig. 6.

E. Faster RCNN

Faster RCNN is a renowned object detection architecture that has significantly influenced the field of computer vision. The architecture comprises convolution layers trained to extract specific features from images, akin to how coffee filters allow only desired elements to pass through.

The Region Proposal Network (RPN) is a pivotal component of Faster RCNN, efficiently generating high-quality region proposals for subsequent detection. Fully connected neural networks are utilized to predict object classes and refine bounding boxes based on the regions proposed by the RPN (see Fig. 7).

Training Faster RCNN involves optimizing convolution layers filters, RPN weights, and the last fully connected layer weights using Stochastic Gradient Descent (SGD), ensuring effective and efficient object detection [26]. Faster RCNN has demonstrated remarkable performance improvements over its predecessors, achieving faster processing speeds during both training and testing phases and setting new standards in object detection accuracy and efficiency.

IV. Experiment

A. Dataset

The "Road Signs" dataset sourced from Roboflow-100 consists of 21 classes falling under the super category "Road Signs". The images sample is shown in Fig. 8.

Fig. 8. Dataset images.

The dataset encompasses around 20 classes. It consists of a total of 2095 images, divided into a training set with 1378 images, a validation set with 488 images, and a test set with 229 images. Preprocessing techniques applied include auto-orientation to adjust image orientation and resizing to a resolution of 640x640 pixels, with no manual augmentations initially applied. However, each object detection model used in the study incorporated its own set of image augmentations during the training process to enhance the training data and improve model robustness and performance. This dataset offers a diverse collection of road sign images across multiple classes, ensuring uniformity in image dimensions and orientation for effective model training and testing.

B. Training Hyperparameters

Yolov8 and Yolov9 were trained using Ultralytics, RetinaNet and FasterRCNN using Detectron2 while RTMdet was trained using MMDetection. In this study, all the models were trained for 30 iterations through the dataset. YOLV8 and YOLV9 were set to 16, while RetinaNet, Faster RCNN, and RTMDet were set to 8. Here, parameters were adjusted for YOLOv8 with AdamW optimizer where the learning rate was set to 0. 0004, momentum of 0.9 and weight decay of 0.0005. As for optimization, YOLOv9 uses Standard Gradient Descent (SGD) with a learning rate of 0.01, momentum of 0.9 and weight decay to 0.0005. YOLOv9 employed SGD with a learning rate of 0.01, momentum of 0.9, and weight decay of 0.0005. RetinaNet and Faster RCNN both used SGD with a learning rate of 0.001, momentum of 0.9, and weight decay of 0.0001. RTMDet was optimized using AdamW with a learning rate of 0.004, momentum of 0.0002, and weight decay of 0.0001. The notes on the training dataset for both YOLOv8 and YOLOv9 were YOLOv5CocoDataset and COCO for RetinaNet, Faster RCNN, and RTMDet.

C. Data Augmentation Techniques

To increase the reliability of object detection models with respect to input images, the concept of data augmentation was used at the time of training. This was achieved by preprocessing the images by using albumentations for YOLOv8, YOLOv9, and RTMDet which applied blur, median blur, combining the image into grayscale, and CLAHE. On the other hand, RetinaNet and Faster RCNN stacked the pre-process tools provided by Detectron2, which involved resizing and random flipping.

D. Evaluation

While evaluating the results of object detection model, COCO evaluation metrics was used to make the evaluation. For instance, in using the model, COCO bounding box(bbox) test methodology was employed to assess the effectiveness of the method in identifying objects and determining the degree of precision. As this evaluation framework was widely employed in determining the effectiveness of such models in detecting malicious data, its use gave us a better chance to arrive at sound conclusions on performance of our powerfully designed model. The classification accuracy are recorded in Fig. 9 to 13.

V. Results and Discussion

A. Mean Average Precision (mAP)

The mean average precision (mAP) is a key metric in evaluating the performance of object detection models. The mAP scores for mAP50-95, mAP50, and mAP75 are shown in Table I.

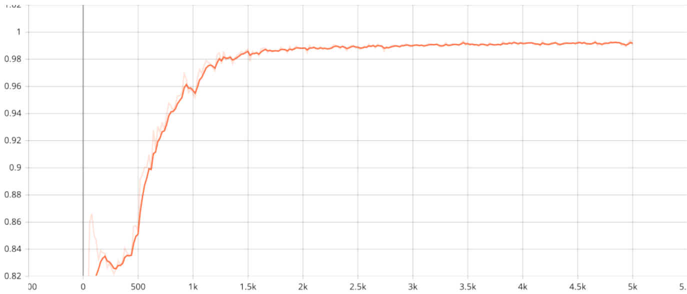

Fig. 9. Faster RCNN: Classification accuracy.

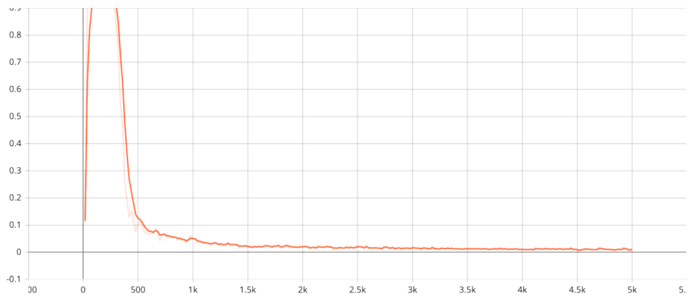

Fig. 10. Faster RCNN: False negatives.

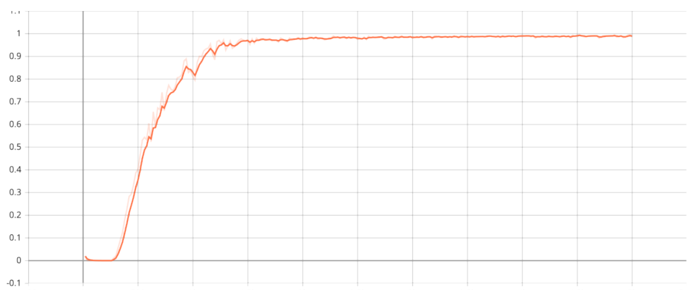

Fig. 11. Faster RCNN: Foreground classification accuracy.

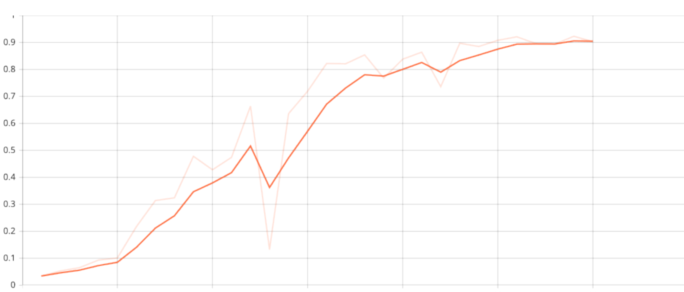

Fig. 12. RTMDet: mAP50.

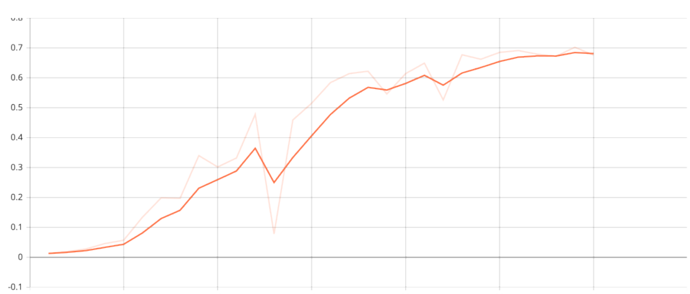

Fig. 13. RTMDet: mAP.

TABLE I. MEAN AVERAGE PRECISION

Model	mAP50-95	mAP50	mAP75
YOLOv8	0.814	0.961	0.892
YOLOv9	0.826	0.973	0.904
RTMDet	0.702	0.923	0.799
Faster RCNN	0.704	0.891	0.816
RetinaNet	0.755	0.911	0.854

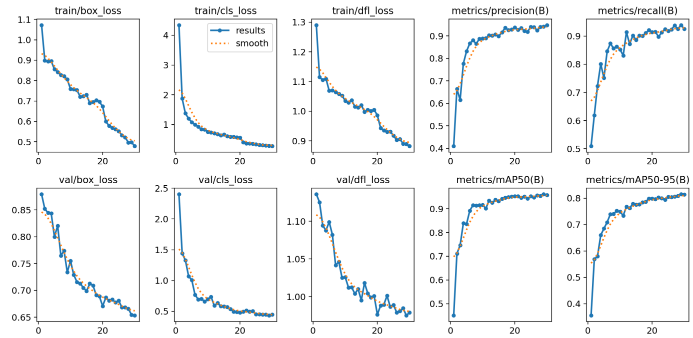

Fig. 14. YOLOv8: Loss and accuracy.

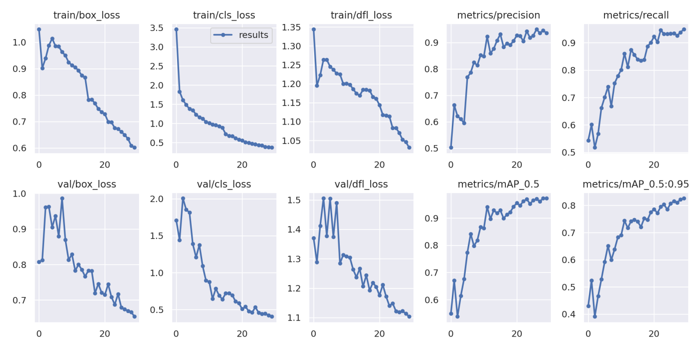

Fig. 15. YOLOv9: Loss and accuracy.

YOLOv9 achieves the highest mAP scores across all metrics, indicating its superior accuracy. YOLOv8 also performs very well, especially in terms of mAP50 and mAP75. RetinaNet shows a balanced performance, while Faster R-CNN and RTMDet exhibit relatively lower mAP scores.

B. *Inference Time*

Inference time is critical for applications requiring real-time object detection. The inference time of these models is shown in Table II.

TABLE II. INFERENCE TIME

Model	Inference Time
YOLOv8	0.0105s
YOLOv9	0.0311s
RTMDet	0.0576s
Faster RCNN	0.0844s
RetinaNet	0.0768s

YOLOv8 is the fastest model, making it highly suitable for real-time applications. YOLOv9, while slower than YOLOv8, still offers reasonable inference time. RTMDet, Faster R-CNN, and RetinaNet are significantly slower, with Faster R-CNN having the highest inference time.

C. Total Loss

Total loss is a metric that indicates the overall error of a model during training, where lower values typically signify better performance. The total loss values for different object detection models: YOLOv8, YOLOv9, RTMDet, Faster RCNN, and RetinaNet, are depicted in Fig. 14 to 18 and presented in Table III. Among the models compared, YOLOv8 and YOLOv9 exhibit higher total loss values, indicating relatively higher error rates during training. On the other hand, RTMDet, Faster RCNN, and RetinaNet demonstrate significantly lower total loss values, suggesting better performance and potentially more effective learning during training.

TABLE III. TOTAL LOSS

Model	Total Loss
YOLOv8	2.0793
YOLOv9	2.1627
RTMDet	0.408
Faster RCNN	0.1519
RetinaNet	0.0735

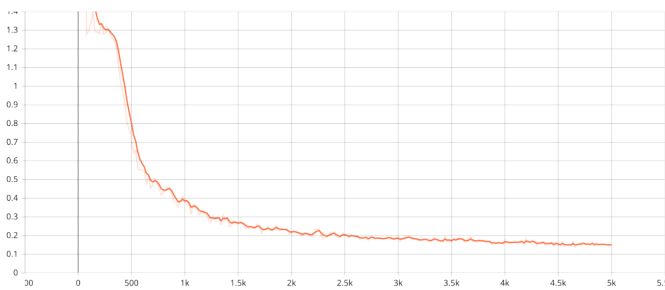

Fig. 16. Faster RCNN: Total loss.

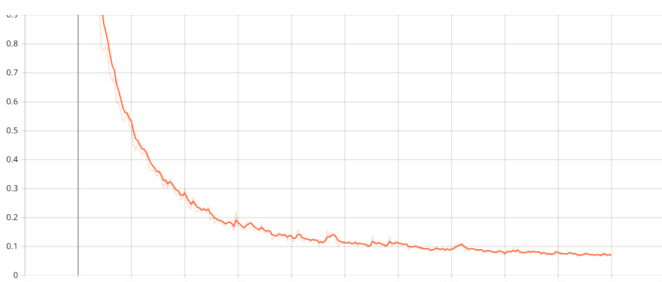

Fig. 17. RetinaNet: Total loss.

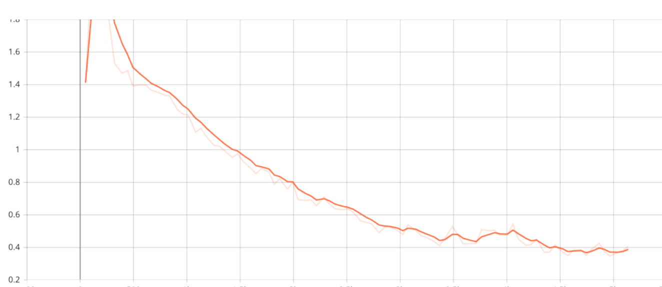

Fig. 18. RTMDet: Total loss.

D. Confusion Matrices

The performance of the YOLOv8, YOLOv9, Faster-RCCN, Retina Net and RTMDet is depicted by confusion matrices in Fig. 18 to 23.

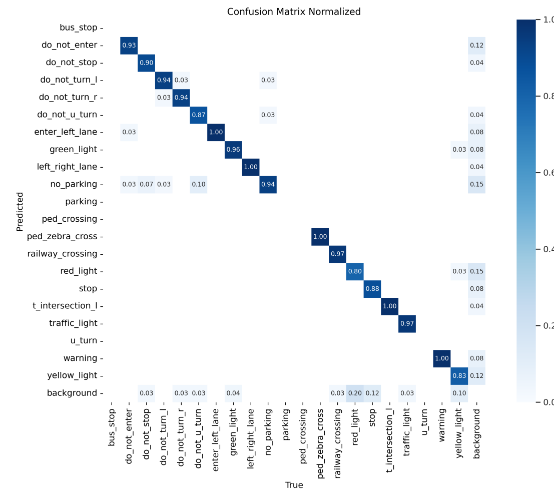

Fig. 19. YOLOv8.

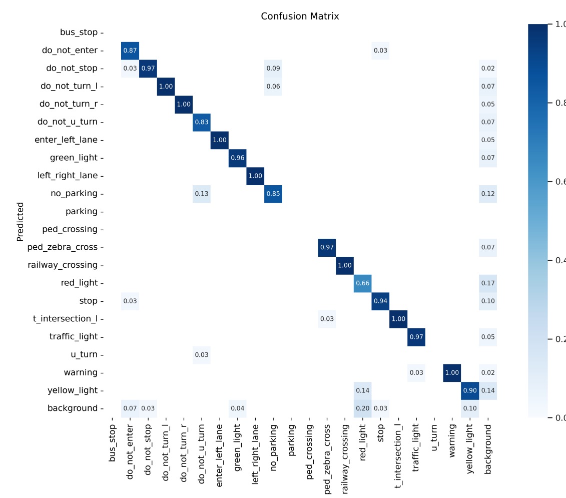

Fig. 20. YOLOv9.

E. Inference Result and Discussion

It is apparent from Fig. 24 to 28, that YOLOv9 followed by YOLOv8 have the highest values of accuracy scores of all the models in the paper. These models are also very accurate especially when it comes to place recognition which is very important with regards to road signs especially for safety reasons. However, on the tradeoff – YOLOv8 is slightly faster on the detection in comparison with YOLOv9. Therefore, if the need is to detect an object in real-time on the road, it may be desirable to use YOLOv8. However, there were other approaches in our experiments, such as RTMDet, Faster RCNN, and RetinaNet, slightly less effective in terms of speed but providing a good enough accuracy. They may be useful when speed is not the major factor into consideration as in case of roads of high importance or, self-driven cars. Besides, in

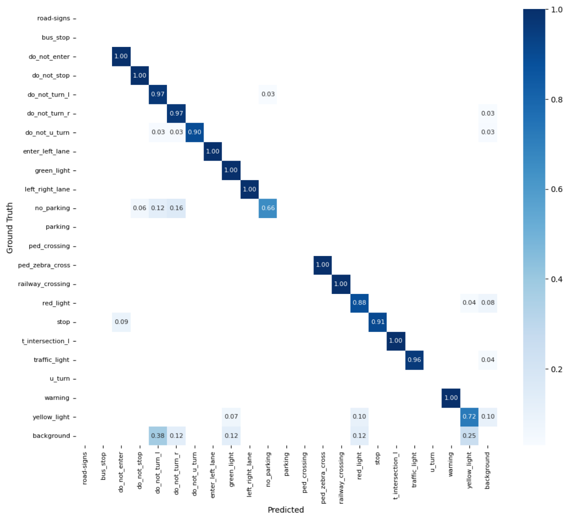

Fig. 21. Faster RCNN.

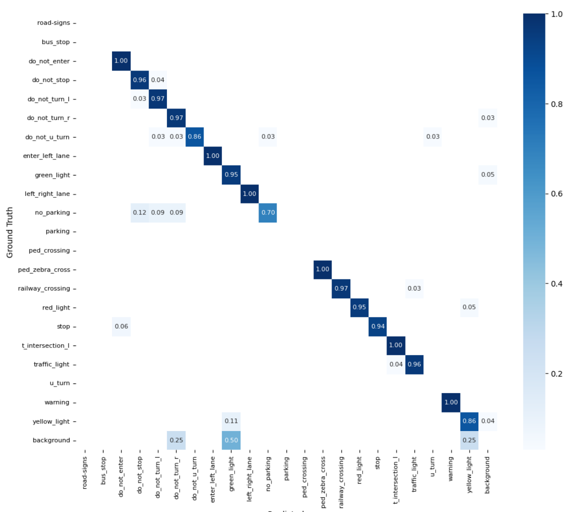

Fig. 22. RetinaNet.

terms of observing the curve of the total loss values, RetinaNet presents the best learning status in training so that it could develop more with more the training dataset. In conclusion, the decision of the specific model to be selected depends with the most valued criterium when it comes to the detection of road signs which can include speed or accuracy or both.

VI. Conclusion

In conclusion, this paper analyzes the performance of different object detection models which include YOLOv8, YOLOv9, RTMDet, RetinaNet and Faster-RCNN for road sign detection. YOLOv9 provides satisfactory results in terms of identification speed and accuracy measure; however, its inference time is relatively longer than the other versions including YOLOv8 with better real-time application performance. RTMDet achieves a better balance which very importantly is essential in scenarios where speed and accuracy are given utmost importance. Faster RCNN and RetinaNet have high enough accuracy, but the time for evaluation takes longer,

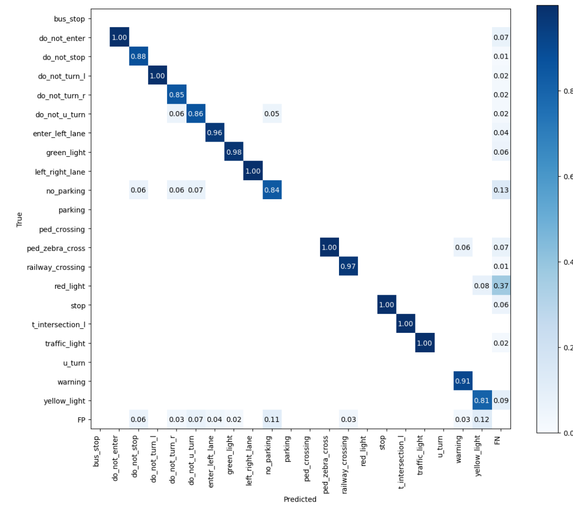

Fig. 23. RTMDet.

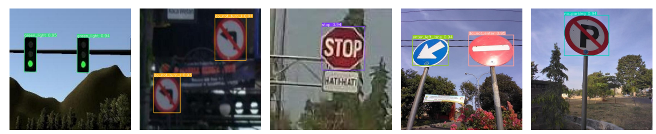

Fig. 24. YOLOv8.

Fig. 25. YOLOv9.

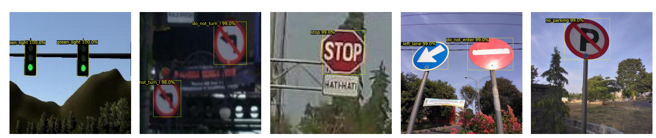

Fig. 26. Faster RCNN.

Fig. 27. RetinaNet.

Fig. 28. RTMDet.

which may be useful for applications requiring a higher probability of correct result. The work underlines the parameters of accuracy, time and confidence in detecting the road contours and signs that can help to choose the best option for the given application and to contribute to the creation of effecting road safety as well as autonomous driving systems. Our future work includes investigating more sophisticated ensemble approaches to achieve high accuracy and avoid overfitting problem, as well as to using more progressive data augmentation methodologies for increasing model's ability to generalize such as generative adversarial network (GAN) or self-supervised learning to real-world conditions and diverse environments, such as varying weather, lighting, and occlusions.

References

[1] A. John and D. Meva, "A comparative study of various object detection algorithms and performance analysis," *International Journal of Computer Sciences and Engineering*, vol. 8, no. 10, pp. 158–163, 2020.

[2] N. Yadav and U. Binay, "Comparative study of object detection algorithms," *International Research Journal of Engineering and Technology (IRJET)*, vol. 4, no. 11, pp. 586–591, 2017.

[3] R. Padilla, W. L. Passos, T. L. Dias, S. L. Netto, and E. A. Da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics*, vol. 10, no. 3, p. 279, 2021.

[4] S. Srivastava, A. V. Divekar, C. Anilkumar, I. Naik, V. Kulkarni, and V. Pattabiraman, "Comparative analysis of deep learning image detection algorithms," *Journal of Big data*, vol. 8, no. 1, p. 66, 2021.

[5] Z. Fatima, S. Zardari, and M. H. Tanveer, "Advancing industrial object detection through domain adaptation: A solution for industry 5.0," *Actuators*, vol. 13, no. 12, 2024. [Online]. Available: https://www.mdpi.com/2076-0825/13/12/513

[6] P. Kaushik, A. Sharma, and B. Kaur, "Comparative analysis of object detection algorithms," in *Advances in Distributed Computing and Machine Learning: Proceedings of ICADCML 2022*. Springer, 2022, pp. 675–685.

[7] M. Soumyadeep, A. P. Singh, A. Sharma, and P. Kumar, "Real-time object detection and tracking using deep learning," in *2023 11th International Conference on Intelligent Systems and Embedded Design (ISED)*. IEEE, 2023, pp. 1–7.

[8] L. Tan, T. Huangfu, L. Wu, and W. Chen, "Comparison of yolo v3, faster r-cnn, and ssd for real-time pill identification," 2021.

[9] J.-H. Hwang, M. Lim, G. Han, H. Park, Y.-B. Kim, J. Park, S.-Y. Jun, J. Lee, and J.-W. Cho, "A comparative study on the implementation of deep learning algorithms for detection of hepatic necrosis in toxicity studies," *Toxicological Research*, vol. 39, no. 3, pp. 399–408, 2023.

[10] J. Wang, S. Jiang, W. Song, and Y. Yang, "A comparative study of small object detection algorithms," in *2019 Chinese control conference (CCC)*. IEEE, 2019, pp. 8507–8512.

[11] T. M. N. B. T. Rashid and L. M. Fadzil, "Comparative review of object detection algorithms in small single-board computers," *International Journal on Recent and Innovation Trends in Computing and Communication*, 2023.

[12] M. Asad, S. Khaliq, M. Yousaf, M. Ullah, and A. Ahmad, "Pothole detection using deep learning: a real-time and ai-on-the-edge perspective. adv civ eng 2022: 1–13," 2022.

[13] K. M. Krishna, A. Sowmya, D. Jerusha, and D. Susmitha, "Comparative study of vehicle detection using ssd and faster rcnn," 2021.

[14] S. Reddy, N. Pillay, and N. Singh, "Comparative study of convolutional neural network object detection algorithms for image processing," in *2023 International Conference on Electrical, Computer and Energy Technologies (ICECET)*. IEEE, 2023, pp. 1–5.

[15] M.-h. Lee and H.-J. Mun, "Comparison analysis and case study for deep learning-based object detection algorithm," *Int. J. Adv. Sci. Converg*, vol. 2, no. 4, pp. 7–16, 2020.

[16] A. K. Shetty, I. Saha, R. M. Sanghvi, S. A. Save, and Y. J. Patel, "A review: Object detection models," in *2021 6th International Conference for Convergence in Technology (I2CT)*. IEEE, 2021, pp. 1–8.

[17] K. Li and L. Cao, "A review of object detection techniques," in *2020 5th International Conference on Electromechanical Control Technology and Transportation (ICECTT)*. IEEE, 2020, pp. 385–390.

[18] F. Sultana, A. Sufian, and P. Dutta, "A review of object detection models based on convolutional neural network," *Intelligent computing: image processing based applications*, pp. 1–16, 2020.

[19] M. H. Tanveer, Z. Fatima, H. Mariam, T. Rehman, and R. C. Voicu, "Three-dimensional outdoor object detection in quadrupedal robots for surveillance navigations," *Actuators*, vol. 13, no. 10, 2024. [Online]. Available: https://www.mdpi.com/2076-0825/13/10/422

[20] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A review of yolo algorithm developments," *Procedia computer science*, vol. 199, pp. 1066–1073, 2022.

[21] G. Lavanya and S. D. Pande, "Enhancing real-time object detection with yolo algorithm," *EAI Endorsed Transactions on Internet of Things*, vol. 10, 2024.

[22] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using yolo: Challenges, architectural successors, datasets and applications," *multimedia Tools and Applications*, vol. 82, no. 6, pp. 9243–9275, 2023.

[23] M. N. Alhasanat, M. H. Alsafasfeh, A. E. Alhasanat, and S. G. Althunibat, "Retinanet-based approach for object detection and distance estimation in an image," *International Journal on Communications Antenna and Propagation (IRECAP)*, vol. 11, no. 1, pp. 1–9, 2021.

[24] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[25] S. Bhatlawande, S. Shilaskar, M. Agrawal, V. Ashtekar, M. Badade, S. Belote, and J. Madake, "Study of object detection with faster rcnn," in *2022 2nd International Conference on Intelligent Technologies (CONIT)*. IEEE, 2022, pp. 1–6.

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[27] C. Lyu, W. Zhang, H. Huang, Y. Zhou, Y. Wang, Y. Liu, S. Zhang, and K. Chen, "Rtmdet: An empirical study of designing real-time object detectors," *arXiv preprint arXiv:2212.07784*, 2022.

[28] Z. Huang, L. Li, G. C. Krizek, and L. Sun, "Research on traffic sign detection based on improved yolov8," *Journal of Computer and Communications*, vol. 11, no. 7, pp. 226–232, 2023.

[29] O. Renuka *et al.*, "A yolov8-based approach for multi-class traffic sign detection," *International Journal of Science and Research Archive*, vol. 11, no. 2, pp. 824–829, 2024.

[30] Y. Zhu and W. Q. Yan, "Traffic sign recognition based on deep learning," *Multimedia Tools and Applications*, vol. 81, no. 13, pp. 17 779–17 791, 2022.

[31] X. Gao, L. Chen, K. Wang, X. Xiong, H. Wang, and Y. Li, "Improved traffic sign detection algorithm based on faster r-cnn," *Applied Sciences*, vol. 12, no. 18, p. 8948, 2022.

[32] E. Güney and C. Bayılmış, "An implementation of traffic signs and road objects detection using faster r-cnn," *Sakarya University Journal of Computer and Information Sciences*, vol. 5, no. 2, pp. 216–224, 2022.

[33] M. Zhu, H. Yang, Z. Cui, and Y. Wang, "Traffic sign detection and recognition for autonomous driving in virtual simulation environment," in *International Conference on Transportation and Development 2022*, 2019, pp. 12–18.

[34] J. Chen, K. Jia, W. Chen, Z. Lv, and R. Zhang, "A real-time and high-precision method for small traffic-signs recognition," *Neural Computing and Applications*, vol. 34, no. 3, pp. 2233–2245, 2022.

[35] B. Senthilnayaki, C. Rajeswary, G. Nivetha, P. Dharanyadevi, G. Mahalakshmi, and A. Devi, "Traffic sign prediction and classification using image processing techniques," in *2022 International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*. IEEE, 2022, pp. 1–5.

[36] A. Aneesh, L. Shine, R. Pradeep, and V. Sajith, "Real-time traffic light detection and recognition based on deep retinanet for self driving cars," in *2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, vol. 1. IEEE, 2019, pp. 1554–1557.

[37] L. Jiang, H. Liu, H. Zhu, and G. Zhang, "Improved yolo v5 with balanced feature pyramid and attention module for traffic sign detection," in *MATEC Web of Conferences*, vol. 355. EDP Sciences, 2022, p. 03023.

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

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

[40] R. Del Prete, M. D. Graziano, and A. Renga, "Retinanet: A deep learning architecture to achieve a robust wake detector in sar images," in *2021 IEEE 6th International Forum on Research and Technology for Society and Industry (RTSI)*. IEEE, 2021, pp. 171–176.