Deep Learning Approach: YOLOv5-based Custom Object Detection

Taoufik Saidani
Department of Computer Sciences, Faculty of Computing and Information Technology, Northern Border University, Rafha, Saudi Arabia | Laboratory of Electronics and Microelectronics (EμE), Faculty of Sciences, Monastir University, Monastir, Tunisia
Taoufik.Saidan@nbu.edu.sa

Received: 14 September 2023 | Revised: 13 October 2023 | Accepted: 16 October 2023

 DOI: https://doi.org/10.48084/etasr.6397

ABSTRACT

Object detection is of significant importance in the field of computer vision, since it has extensive applications across many sectors. The emergence of YOLO (You Only Look Once) has brought about substantial changes in this domain with the introduction of real-time object identification with exceptional accuracy. The YOLOv5 architecture is highly sought after because of its increased flexibility and computational efficiency. This research provides an in-depth analysis of implementing YOLOv5 for object identification. This research delves deeply into the architectural improvements and design ideas that set YOLOv5 apart from its predecessors to illuminate its unique benefits. This research examines the training process and the efficiency of transfer learning techniques, among other things. The detection skills of YOLOv5 may be greatly improved by including these features. This study suggests the use of YOLOv5, a state-of-the-art object identification framework, as a crucial tool in the field of computer vision for accurate object recognition. The results of the proposed framework demonstrate higher performance in terms of mAP (60.9%) when evaluated with an IoU criterion of 0.5 and when compared to current methodologies in terms of reliability, computing flexibility, and mean average precision. These advantages make it applicable in many real-world circumstances.

Keywords-computer vision; object detection; deep learning; YOLOv5

I. INTRODUCTION

The identification and localization of objects inside pictures or videos is a crucial challenge in the field of computer vision. The subject matter has garnered considerable interest as a result of its extensive array of practical uses, including autonomous vehicular navigation, surveillance technology, robotic systems, and augmented reality experiences. Numerous object identification algorithms encounter challenges in properly detecting tiny items while demonstrating satisfactory performance on bigger objects. Small objects are entities that possess a limited pixel area or field of vision within an input picture. The aforementioned concern emerges due to the tendency of generic object detectors to assign more priority to the characteristics of bigger objects when they traverse numerous levels of their underlying architecture. The identification of diminutive entities poses several obstacles, such as subpar visual characteristics, restricted contextual data, noisy depiction, indiscernible attributes, intricate backdrops, restricted resolution, substantial occlusion, etc. [1]. Real-time object detection systems often favor computing speed above resource utilization. However, their performance in terms of detection accuracy is generally subpar, making them unsuitable for many practical uses. In the domain of autonomous driving, the detection of objects on the roadway is an essential undertaking. Conventional road object detection systems often demonstrate less accuracy in detecting tiny things. This stems from the fact that smaller objects tend to occupy a smaller number of pixels, hence posing difficulties in extracting significant information from representations with low resolution. As a result, the models have a tendency to erroneously classify tiny items as part of the background, resulting in failure to identify or incorrectly detect such things [2]. In addition, the task of reliably identifying things of varying sizes is a substantial obstacle for object detection algorithms. In the context of autonomous driving, little items include road signs and traffic signals. It has been proposed that enhancing the network's representational capacity via increased depth and breadth may lead to improved accuracy. However, it is important to note that this strategy also results in increased model complexity and cost. Consequently, these models are not well suited for self-driving cars with real-time requirements and limited resources. The models used by deep learning for object identification may be classified into two main categories: two-stage and one-stage detection techniques [3]. Two-stage models provide superior accuracy, at the cost of reduced speed and increased complexity, hence diminishing their practicality in driving circumstances. Current research has been dedicated to enhancing the performance of one-stage models [1, 3], resulting in the creation of many novel one-stage detectors specifically suited for real-world usage.

The work at hand focuses on a widely used one-stage object detection model called You Only Look Once version 5 (YOLOv5) [4]. YOLO is regarded as a very significant and extensively used method for object identification. The YOLO algorithm has significantly advanced the field of object detection by approaching it as a cohesive job, hence facilitating real-time inference while achieving remarkable levels of accuracy, with YOLOv5 being the latest iteration within the YOLO lineage, distinguished by its well-defined and adaptable architecture, with the primary objective of attaining superior performance and rapidity on widely available platforms. Nevertheless, the systems that use YOLOv5 mostly depend on traditional training methodologies, regularization, and normalization approaches, or parameter tuning to improve performance, often neglecting architectural alterations. Although YOLOv5 serves as a versatile object detector, its optimization does not especially cater to the identification of tiny objects. Consequently, its applicability to real scenarios is limited [5].

This research presents a comprehensive analysis of the architectural components of YOLOv5, specifically focusing on its backbone, neck, and head networks. These networks collaborate harmoniously to facilitate the accurate identification and spatial positioning of objects. In addition, we examine the training procedure of YOLOv5, which includes essential stages such as data preprocessing, model refinement, and hyperparameter adjustment. Furthermore, an examination is conducted on the primary methodologies used in the process of inference, encompassing bounding box forecasting and post-processing strategies such as non-maximum suppression. Through a complete examination of these characteristics, our objective is to provide a clear knowledge of the internal mechanisms of YOLOv5 and its fundamental elements in the context of object detection.

II. PROPOSED YOLOV5-BASED OBJECT DETECTION

There is an increasing demand for artificially intelligent systems able to conduct real-time object identification and recognition on resource-constrained devices in the age of IoT (Internet of Things) and edge computing. Our research is motivated by the actual applications of such technology, which include surveillance, autonomous robots, and a variety of other scenarios in which embedded devices must make intelligent judgments based on visual data. On the other hand, embedded devices frequently have constrained processing resources, memory, and power consumption. Driven by such limitations, we intend to create an effective object identification method capable of operating successfully within these limits. The primary emphasis of our study is centered on the training and inference of object detection using the YOLOv5 tiny model. The problem of object detection has significant importance in the field of computer vision, since tailoring the model to particular items or domains may significantly improve its performance and practicality in real-world situations. We used the YOLOv5 tiny model (Figure 1), which presents a harmonious trade-off between the dimensions of the model and its precision. This condensed iteration is appropriate for situations with limited resources or applications that need real-time processing.

During the training phase, the tailored strategy is used. We compiled a dataset including distinct items that are pertinent to our designated application. The dataset potentially comprises of annotated photos or videos, whereby bounding box annotations are used to indicate the positions of objects. We strive to provide a sufficient quantity of examples that include a wide range of object modifications, orientations, and backgrounds. Subsequently, the YOLOv5 tiny model is subjected to fine-tuning using our customized dataset. The process of fine-tuning entails the initialization of the model with pre-existing weights, followed by training it on our dataset to acquire knowledge unique to the item identification task at hand. Transfer learning methods are used in order to enhance the training process and enhance the performance of the model by using the information acquired during pre-training on a substantial dataset. Throughout the training procedure, hyperparameters, including learning rate, batch size, and regularization approaches, are fine-tuned in order to get optimal outcome. In addition, data augmentation methods, including random cropping, rotation, and scaling, were used to enhance the resilience and variety of the training samples.

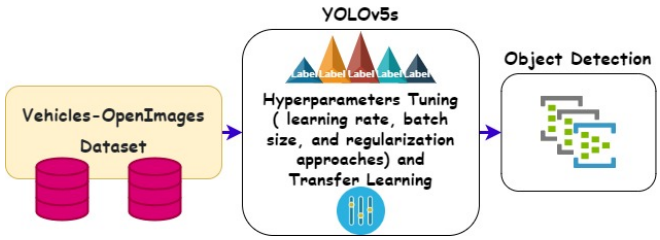

Fig. 1. The proposed object detection framework.

After the completion of the training process, the subsequent step involves transitioning to the inference phase. During this stage, the trained YOLOv5 tiny model is used to identify items inside novel and unobserved photos or videos. The model undertakes the processing of the incoming data and produces predictions of bounding boxes, along with the appropriate probability of class membership for each discovered item. Techniques such as non-maximum suppression are used to eliminate overlapping or duplicate detections, therefore preserving just the most confident and accurate ones. The use of our proprietary object identification technique using the YOLOv5 tiny model has several benefits. By customizing the model to our particular targets of interest, we may enhance the precision of detection and improve its ability to apply to real-world situations. Furthermore, the tiny dimensions of YOLOv5 facilitate effective implementation on edge devices or when the computing resources are restricted. In summary, this study showcases the efficacy of object detection training and inference using the tiny YOLOv5 model, enabling researchers and practitioners to effectively tackle object detection difficulties and devise precise and efficient solutions.

A. YOLOv5 Tiny

YOLOv5 tiny is a modified version of the YOLOv5 object detection model that provides a trade-off between the size of the model and its detection capabilities. The primary objective of its design is to provide a comparatively more streamlined and effective solution in contrast to the bulkier iterations of YOLOv5, while also maintaining commendable levels of detection accuracy. The term "tiny" pertains to its diminished model size, which is accomplished by using diverse architectural optimizations and network scaling strategies. The objective of these improvements is to reduce the quantity of parameters and computing demands of the model, rendering it more suited for deployment on devices with limited resources or situations where real-time performance is of utmost importance [4]. Despite its reduced dimensions, YOLOv5 tiny effectively preserves the essential components and concepts inherent in the YOLOv5 design. The system consists of a primary network, an intermediate network, and a final network, collaborating harmoniously to identify and determine the precise location of objects inside photos or videos. The backbone network is responsible for extracting high-level characteristics from the input data. These features are then further refined by the neck network. Finally, the head network utilizes these refined features to forecast bounding boxes and class probabilities for the identified objects.

The process of training YOLOv5 tiny includes data preparation, model optimization, and hyperparameter tweaking, which aligns with the methodology used in other YOLOv5 versions. In order to do object detection, it is customary to use a dataset that is labeled, consisting of photos or videos that have been tagged with bounding boxes denoting the positions of the objects. Through the process of training on this dataset, YOLOv5 tiny acquires the ability to discover and categorize things that are of significance. During the process of inference, YOLOv5 tiny employs many strategies like bounding box prediction and post-processing processes such as non-maximum suppression in order to provide object detections that are both accurate and dependable. The system performs real-time processing of incoming photos or videos, generating predictions of bounding boxes and corresponding probability for the identified items.

In general, YOLOv5 tiny offers a somewhat lighter and more efficient alternative for object detection assignments when compared to its bigger counterparts within the YOLOv5 framework. This technology is especially well-suited for use cases in which there are constraints on processing resources or a need for real-time performance, all while maintaining a high level of detection accuracy. In this study, after fine-tuning the model to process the custom dataset, the YOLOv5s contains 214 layers, 7033114 parameters and gradients, and 16 GFLOPs.

B. Dataset

The Vehicles-OpenImages dataset [6] is a publicly accessible dataset that has been deliberately curated to concentrate on cars. The subset in question is derived from the dataset, a comprehensive collection of images that includes annotations for a diverse range of item classifications. It comprises a comprehensive assortment of photographs showcasing a vast array of vehicles, including various types such as automobiles, trucks, motorbikes, bicycles, and other similar modes of transportation. The collection has annotated bounding boxes that provide information about the precise positioning of automobiles inside each picture.

The inclusion of annotations inside the Vehicles-OpenImages dataset is a valuable resource for academics and developers, as it allows them to effectively train and assess object identification models that are especially designed for the purpose of detecting vehicles. By using this dataset, professionals may construct and optimize models that possess the ability to precisely identify and determine the location of cars in various situations and circumstances. The accessibility of the Vehicles-OpenImages collection enables progress in domains such as autonomous vehicle technology, traffic surveillance, and transportation studies. This dataset may be used to train vehicle detection models that are resilient, hence facilitating the development of applications that depend on precise and effective vehicle identification and tracking.

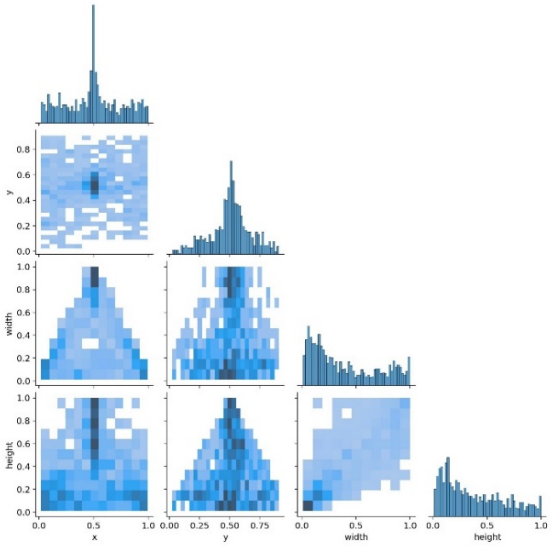

Fig. 2. Dataset label correlogram.

The correlogram label, as seen in Figure 2, is a visualization method used for examining the interrelationships among labels or classes inside a multi-label categorization scenario. This analysis offers valuable observations into the patterns of co-occurrence and interdependencies among various labels present in a given dataset. This Figure depicts the representation of labels or classes on the x and y axes of a matrix or heatmap. Every individual element inside the matrix denotes the correlation or co-occurrence between a certain pair of labels. The calculation of correlation is often performed with statistical metrics such as mutual information, Jaccard index, or correlation coefficient. The intensity or hue of each cell denotes the magnitude of the correlation between the labels. The presence of a high-intensity or dark-colored cell indicates a robust positive correlation, signifying a high likelihood of co-occurrence between the associated labels. On the contrary, a cell with low intensity or light coloration signifies a diminished

or adverse correlation, suggesting a reduced likelihood of co-occurrence between the five chosen labels (ambulance, bus, car, motorcycle, and truck).

Figure 3(a) depicts a graphical representation illustrating the frequency of annotations per class within the dataset. Figure 3(b) illustrates the spatial distribution and dimensions of the bounding boxes within the dataset. This analysis aids in comprehending the distribution patterns of the bounding boxes within the dataset, so ensuring the presence of enough diversity in object location and size for effective recognition by the model. Figures 3(c)-(d) depict the statistical distribution of the location and the size of the bounding boxes, respectively, providing insight of the distribution patterns of the bounding boxes throughout the dataset. This graph facilitates the assessment of the distribution of bounding boxes, indicating if there is an equal distribution or whether some regions of the dataset exhibit a higher concentration. It is crucial to ensure that the model is capable of accurately recognizing items inside a picture, since objects exhibit variations in both size and location. Figure 4 represents a sample of the dataset that contains 627 images divided into five classes.

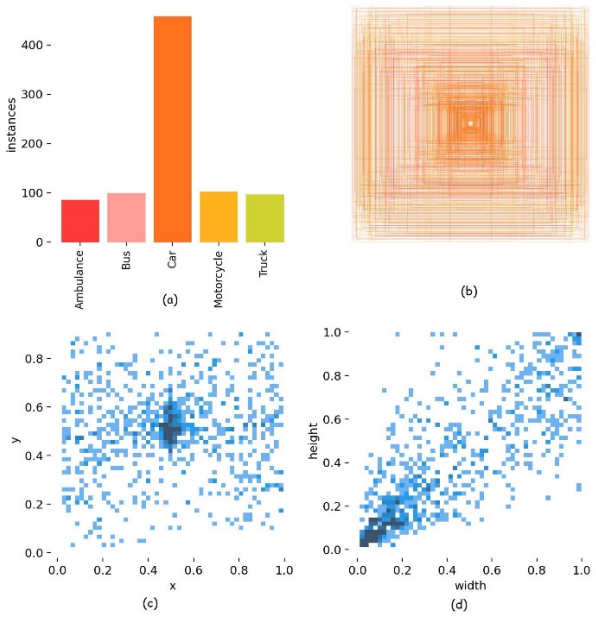

Fig. 3. (a) Dataset's graphical representation of the number of annotations per class, (b) size and location of each bounding box, (c) statistical distribution of the bounding box position, and (d) statistical distribution of the bounding box size.

III. RESULTS AND DISCUSSION

A. Evaluation Metrics

Model assessment is significant in evaluating its performance and its alignment with the research goals. Various assessment metrics may be used in the context of object identification, including criteria pertaining to the precision, swiftness, and efficacy of the model. The primary emphasis on recognizing objects often is on evaluation metrics pertaining to accuracy, since it is essential for the model to possess the capability of accurately identifying buses. Precision (P) and recall (R) are often used as fundamental assessment criteria [7,8]. R quantifies the model's ability to accurately identify positive classifications. In addition to P and R, there are other assessment metrics that are often used in the context of object identification, namely Average Precision (AP) and mean Average Precision (mAP). The AP metric quantifies the model's ability to accurately identify pertinent things while excluding extraneous ones. It is determined by graphing the precision-recall curve and computing the area under the curve. mAP, conversely, is the arithmetic mean of the APs calculated for each individual object class detected by the model [10], whereas mAP offers a more thorough assessment of the model's performance since it evaluates the accuracy across all object classes, rather than focusing just on a single class. These metrics are described by the following equations:

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \quad (1)$$

$$\mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \quad (2)$$

$$\mathrm{AP} = \int_0^1 P(r)\, dr \quad (3)$$

$$\mathrm{mAP} = \frac{\sum_{i=1}^{k} \mathrm{AP}_i}{k} \quad (4)$$

where TP stands for True Positive, FP for False Positive, and FN for False Negative.

Fig. 4. Dataset sample.

B. Training Results

The training process of the proposed framework for custom object detection employs a stochastic gradient descent with learning rate of 0.01, batch size of 16, 100 epochs, 439 images for the training set, and 125 images for the validation set. Figure 5 illustrates the training performances of the proposed framework, which is based on YOLOv5s. Despite the YOLOv5s model's parameter count of around 7 million, the achieved outcomes demonstrate a high level of efficiency. The whole training procedure was accomplished within a time frame of around 0.41 hours, using a mid-range GPU. The YOLOv5s model demonstrated a mAP of 60.9% when

evaluated using an IoU criterion of 0.5. This finding suggests that the model has the capability to effectively identify and pinpoint objects with a notable level of accuracy. Furthermore, when considering a broader spectrum of IoU thresholds ranging from 0.5 to 0.95, the model exhibited mAP values of 44%. This indicates that the model has the capability to sustain a relatively elevated level of performance even when confronted with diverse degrees of overlap between predicted and actual bounding boxes. The aforementioned findings underscore the efficacy and proficiency of the suggested framework using YOLOv5s for the purpose of object detection assignments. Despite its comparatively smaller dimensions in relation to bigger YOLOv5 variations, the model exhibits robust performance, making it well-suited for real-time applications or situations characterized by constrained computing resources.

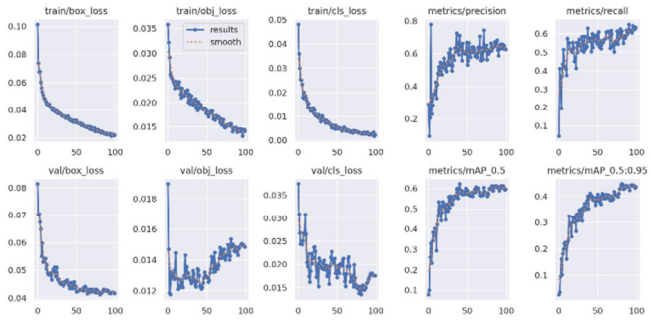

Fig. 5. Loss and mAP results.

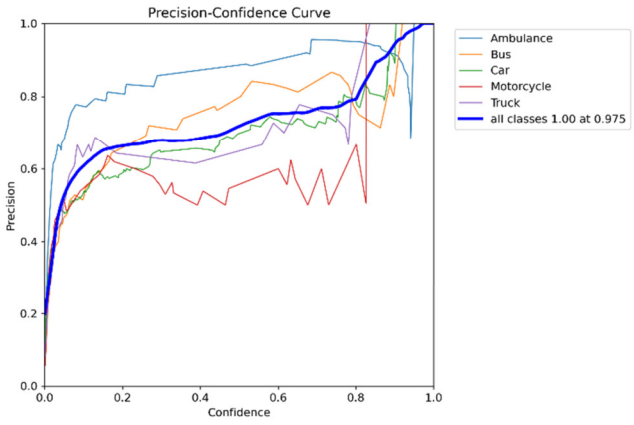

Fig. 6. Precision-condifence results.

The precision-confidence results, as indicated in Figure. 6, pertain to the evaluation of the accuracy of object identification predictions across various confidence levels. When assessing object detection models, it is customary to examine several confidence criteria in order to ascertain the accuracy of the detected objects. In this particular instance, the precision-confidence outcomes reveal that every class attained a precision value of 1 (100%) when subjected to a confidence threshold of 0.975, which is considered to be high. This indicates that all predictions made by the method with a confidence greater than 0.975 were either accurate or qualified as genuine positives. For a given confidence level, a precision value of 1 means that no FP detections occurred for any of the classes. The model is doing well and can be trusted to make correct predictions when the accuracy metric displays a high value and the confidence threshold is set to a high level. In situations where a low rate of false positives is critical, this indicates that the model can give highly trustworthy detections.

By analyzing the precision-recall curve, in Figure 7, it is important to evaluate the effectiveness of the model by considering its performance at different levels of confidence thresholds. A greater level of accuracy is indicative of a reduced occurrence of false positives, but a higher recall value implies a decreased occurrence of false negatives. The determination of the ideal trade-off between accuracy and recall is contingent upon the unique demands and preferences of the given application, and may be ascertained by an examination of the form and attributes of the precision-recall curve.

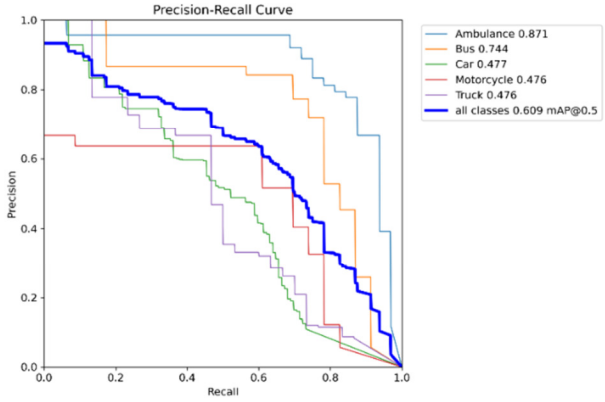

Fig. 7. Precision-recall results.

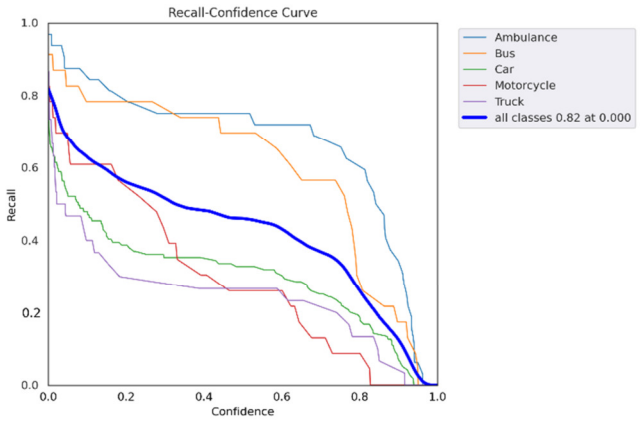

Fig. 8. Recall-confidence results.

Figure 8 displays the results of the analysis of the relationship between prediction confidence and recall for item identification. The proportion of really positive detections that the model gets right is measured by the recall metric. All classes were able to reach a memory rate of 0.82 (82%) with a

confidence threshold of 0.0, as shown by the presented recall-confidence results. In other words, the model correctly identified TP detections across all classes 82% of the time, without using a confidence threshold. The recall-confidence curve may be used to assess the model's accuracy at different levels of certainty. This helps evaluate how well various levels of confidence are reflected in the trade-off between the number of correctly recognized objects (recall) and the prevalence of false positive identifications (precision). The outcomes of detection of the proposed model are shown in Figure. 9.

Fig. 9. Detection results.

IV. CONCLUSION

This paper presents the development of a customized object detection model using the YOLOv5 architecture, with a special focus on addressing the unique issues encountered in the field of autonomous driving. The main objective of our study was to develop a model that is both lightweight and computationally efficient, while maintaining a high level of accuracy. The use of the YOLOv5 architecture allowed us to get optimal outcomes in relation to the dimensions of the model and the computing demands. The architectural design of YOLOv5 achieves a harmonious equilibrium between the intricacy of the model and its ability to accurately recognize objects, rendering it well-suited for deployment in situations with limited computational resources, such as autonomous driving systems. The proposed object identification model exhibited a notable level of precision in identifying and precisely locating things that are pertinent to the activities associated with autonomous driving. The efficacy of the model in carrying out these crucial duties was confirmed by a thorough assessment process. The inherent property of our model being lightweight facilitates expedited inference times, a critical need in time-sensitive applications like autonomous driving.

The current study's shortcomings include the need for additional varied datasets, space limits on full comparison research, and the continued difficulty of meeting real-time needs while managing speed and precision.

REFERENCES

[1] W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling, and R. Yang, "Salient Object Detection in the Deep Learning Era: An In-Depth Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 6, pp. 3239–3259, Jun. 2022, https://doi.org/10.1109/TPAMI.2021.3051099.

[2] I. Bilik, O. Longman, S. Villeval, and J. Tabrikian, "The Rise of Radar for Autonomous Vehicles: Signal Processing Solutions and Future Research Directions," *IEEE Signal Processing Magazine*, vol. 36, no. 5, pp. 20–31, Sep. 2019, https://doi.org/10.1109/MSP.2019.2926573.

[3] F. Guede-Fernández, L. Martins, R. V. de Almeida, H. Gamboa, and P. Vieira, "A Deep Learning Based Object Identification System for Forest Fire Detection," *Fire*, vol. 4, no. 4, Dec. 2021, Art. no. 75, https://doi.org/10.3390/fire4040075.

[4] Y. Zhang, Z. Guo, J. Wu, Y. Tian, H. Tang, and X. Guo, "Real-Time Vehicle Detection Based on Improved YOLO v5," *Sustainability*, vol. 14, no. 19, Jan. 2022, Art. no. 12274, https://doi.org/10.3390/su141912274.

[5] B. Mahaur and K. K. Mishra, "Small-object detection based on YOLOv5 in autonomous driving systems," *Pattern Recognition Letters*, vol. 168, pp. 115–122, Apr. 2023, https://doi.org/10.1016/j.patrec.2023.03.009.

[6] J. Solawetz, "Vehicles-OpenImages Dataset," *Roboflow*. https://public.roboflow.com/object-detection/vehicles-openimages.

[7] R. Arifando, S. Eto, and C. Wada, "Improved YOLOv5-Based Lightweight Object Detection Algorithm for People with Visual Impairment to Detect Buses," *Applied Sciences*, vol. 13, no. 9, Jan. 2023, Art. no. 5802, https://doi.org/10.3390/app13095802.

[8] M. Salemdeeb and S. Erturk, "Multi-national and Multi-language License Plate Detection using Convolutional Neural Networks," *Engineering, Technology & Applied Science Research*, vol. 10, no. 4, pp. 5979–5985, Aug. 2020, https://doi.org/10.48084/etasr.3573.

[9] S. Sahel, M. Alsahafi, M. Alghamdi, and T. Alsubait, "Logo Detection Using Deep Learning with Pretrained CNN Models," *Engineering, Technology & Applied Science Research*, vol. 11, no. 1, pp. 6724–6729, Feb. 2021, https://doi.org/10.48084/etasr.3919.

[10] D. D. Van, "Application of Advanced Deep Convolutional Neural Networks for the Recognition of Road Surface Anomalies," *Engineering, Technology & Applied Science Research*, vol. 13, no. 3, pp. 10765–10768, Jun. 2023, https://doi.org/10.48084/etasr.5890.

[11] M.-F. R. Lee and Y.-C. Chen, "Artificial Intelligence Based Object Detection and Tracking for a Small Underwater Robot," *Processes*, vol. 11, no. 2, Feb. 2023, Art. no. 312, https://doi.org/10.3390/pr11020312.

[12] J. Li *et al.*, "Detection of Smoke from Straw Burning Using Sentinel-2 Satellite Data and an Improved YOLOv5s Algorithm," *Remote Sensing*, vol. 15, no. 10, Jan. 2023, Art. no. 2641, https://doi.org/10.3390/rs15102641.