Pattern Recognition Letters 168 (2023) 115–122

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Small-object detection based on YOLOv5 in autonomous driving systems

Bharat Mahaur*, K.K. Mishra

Department of Computer Science and Engineering, Motilal Nehru National Institute of Technology Allahabad, UP, India

ARTICLE INFO

Article history:
Received 12 July 2022
Revised 29 January 2023
Accepted 5 March 2023
Available online 7 March 2023

Edited by: Prof. S. Sarkar

Keywords:
Architectural changes
Deep learning
Autonomous driving
Small object detection
YOLOv5

ABSTRACT

With the rapid advancements in the field of autonomous driving, the need for faster and more accurate object detection frameworks has become a necessity. Many recent deep learning-based object detectors have shown compelling performance for the detection of large objects in a variety of real-time driving applications. However, the detection of small objects such as traffic signs and traffic lights is a challenging task owing to the complex nature of such objects. Additionally, the complexity present in a few images due to the existence of foreground/background imbalance and perspective distortion caused by adverse weather and low-lighting conditions further makes it difficult to detect small objects accurately. In this letter, we investigate how an existing object detector can be adjusted to address specific tasks and how these modifications can impact the detection of small objects. To achieve this, we explore and introduce architectural changes to the popular YOLOv5 model to improve its performance in the detection of small objects without sacrificing the detection accuracy of large objects, particularly in autonomous driving. We will show that our modifications barely increase the computational complexity but significantly improve the detection accuracy and speed. Compared to the conventional YOLOv5, the proposed iS-YOLOv5 model increases the mean Average Precision (mAP) by 3.35% on the BDD100K dataset. Nevertheless, our proposed model improves the detection speed by 2.57 frames per second (FPS) compared to the YOLOv5 model.

1. Introduction

Object detection is a very fundamental and well-studied task in the community of computer vision. The purpose of the object detection task is to classify and localize target objects in an image. With the recent advancements in deep learning technologies over the years, several state-of-the-art methods for object detection have emerged [1]. Object detection has been widely applied to many real-world applications, including autonomous driving, robot vision, intelligent transportation, remote sensing, military operations, and surveillance [2–4].

Several object detectors usually perform well on large objects but poorly on small objects. We refer to objects as small when their occupying pixel area or field of view is small in an input image. In the case of generic object detectors, the features of small objects lose importance as they are processed through multiple layers of their backbone. The accurate detection of small objects is indispensable and challenging due to poor visual appearance, insufficient context information, noisy representation, indistinguishable features, complicated backgrounds, limited resolution, severe occlusion, etc [5,6]. Although modern systems designed to implement object detection in real-time mainly focus on speed at the cost of computational resources, they lack feasibility due to their poor detection accuracy. Thus, improvements in this specific area would benefit practical implications in autonomous driving systems.

Detecting target objects on the road is an essential task for autonomous driving. For most existing road object detectors, the detection accuracy for small objects is less than half that of large objects. This is because they usually cover fewer pixels, and it is difficult to extract features from low resolution, so the model can easily confuse it with the background, resulting in missed or incorrect detection [5]. Moreover, one of the most critical challenges of an object detector is that the accurate detection of different scale objects is not well-balanced. In the context of autonomous driving, traffic signs and traffic lights can be regarded as small objects. Although many studies [7,8] suggest increasing the representational capacity of the network in terms of depth and width for accurate detection, this impacts the complexity and cost of the model. Accordingly, such models are less suited for

* Corresponding author.
E-mail addresses: bharatmahaur@gmail.com (B. Mahaur), kkm@mnnit.ac.in (K.K. Mishra).

https://doi.org/10.1016/j.patrec.2023.03.009
0167-8655/

autonomous driving systems because of their real-time resource constraints.

In general, deep learning-based object detection models are categorized into (1) Two-stage detection algorithms and (2) One-stage detection algorithms [1,3]. The two-stage models achieve higher accuracy than one-stage models at the cost of speed and complexity but may not directly benefit the practical driving scenarios. Recently, efforts have been put to match or even improve the performance of one-stage models [2,4]. Hence, many new one-stage detectors have been developed for such applications. In this letter, we focus on the popular one-stage detector, i.e., You Only Look Once version 5 (YOLOv5) model [9]. This is the most recent version in the YOLO family with a clear and flexible structure aiming for high performance and speed on accessible platforms. However, the current systems that apply this model rely either on conventional training methods, regularization/normalization techniques, or adjusting specific parameters to improve performance, with limited or no consideration for architectural modifications. Although YOLOv5 is a generic object detector, it is not optimized for the detection of small objects, and therefore cannot adapt to specific use cases in practice.

This letter proposes architectural improvements to the original YOLOv5 model to perform better in terms of small object detection. For this, we consider the actual road environment in autonomous driving systems to detect small road objects like traffic signs and traffic lights. Moreover, we will discuss the effects of our modifications on how to accurately perform this task while maintaining real-time speed and with a slight increase in the computational complexity of the system. The highlights of our contributions are

- We optimize the existing YOLOv5 model and design a modified YOLOv5 architecture, with the name iS-YOLOv5, aiming for better detection of small objects in autonomous driving scenarios.
- We investigate the applicability of our model in diverse weather scenarios to highlight its significance in the context of more robust and efficient object detection.
- Extensive experimentation on BDD100K dataset demonstrates the efficacy of the proposed model. Moreover, we provide empirical results for traffic sign and traffic light detection on TT100K and DTLD datasets, respectively.

2. Related work

Over the years, many researchers have shown a significant interest in developing and employing deep learning-based models for performance enhancement in object detection tasks. With the advent of the YOLO series [6,10], various applications have utilized YOLO and its architectural successors for object detection due to their real-time detection speed rather than considering detection accuracy. Hence, many investigative studies have proposed applying the YOLO models in autonomous driving scenarios for detecting road objects. For instance, the works [11–13] implemented YOLO v1-v3 for real-time object detection for autonomous vehicles in clear environments. Similarly, [14,15] exploited the benefits of YOLOv4 for the detection of specific road objects in ideal scenarios. All of these methods achieve promising results but do not make an effort to modify the architecture. Besides, [16,17] explored the advantages of structural changes to YOLOv4 for improving detection accuracy in limited driving scenarios. However, these approaches are not universally applicable because they do not account for increasing complexity and inference time. Although the overall structure of YOLOv4 is similar to that of YOLOv5, the latter is focused on accessibility and is mainly designed for low-computing platforms [9,18]. Additionally, systems that use YOLOv5 benefit from less complexity and balanced performance in a variety of practical applications.

Recently, some effort has been made into adjusting and implementing the YOLO models for the detection of small objects in autonomous driving scenarios. For instance, [19–23] exploited the benefits of YOLO v3-v4 for the detection of road objects like traffic signs and/or traffic lights. Some works [24–26] implemented YOLOv5 for either traffic sign or traffic light detection. Similarly, [27,28] explored the benefits of YOLO v4-v5 for traffic cone detection. All of these works attempted to optimize the YOLO models, but only to a limited extent since typical changes to their original structure are common. Moreover, [29–32] combined additional modules to the YOLOv5 feature extractor for refining the detection of small-scale objects. However, such methods mainly focus on inference speed, sacrificing detection accuracy at the cost of more resources, if needed. Besides, structural modifications have proven to be relatively more effective than other techniques for improving small object detection performance [27,33]. In addition, the aforementioned works showed detection results for small objects in plain weather conditions. Because an autonomous vehicle on the road encounters a variety of road environments, analysis made during the daytime will not suffice in challenging weather scenarios. More importantly, all of the current YOLOv5-based systems fail to measure computational cost, which is an important metric for autonomous driving. Also, in most cases [22,26,30], the performance improvements in small object detection tasks are triggered by regularization and normalization schemes. Therefore, in this letter, we investigate how to improve detection performance, especially for small objects, at minimal cost through architectural changes, without using any additional techniques.

3. Methodology

We first discuss the motivations of our work (Section 3.1). Then, we provide a brief overview of YOLOv5 architecture and discuss its shortcomings (Section 3.2). Finally, a series of novel architectural changes are introduced to optimize and improve the detection performance of small objects (Section 3.3).

3.1. Motivations

Although some techniques [13,15,22] have been developed for enhancing the performance of small object detection, only a handful of researchers focus on architectural modifications [24,33] for achieving the same. In most cases, the gain in performance is mainly driven by additional regularization/normalization approaches [26,32] or by increasing the parameters in the framework [8,16]. Hence, it is unclear how architectural refinements contribute towards improving the detection performance for explicit tasks. In autonomous driving, accurate detection of small objects provides more valuable contextual information about the environment that can help better decision-making strategies. The detection of small objects is more challenging because of foreground/background imbalance, fewer appearance cues, and lower image cover rate [3,6]. In a typical road traffic environment, detecting small objects is considered a hard problem as distant objects become smaller due to perspective distortion [34]. Notably, there are significant differences between the localization of small objects of different sizes, even in the same class. Also, many driving systems prioritize inference time over performance, but there are workarounds to optimize them at low cost. Consequently, a simple and efficient road object detection model that can handle small and large objects of different sizes is needed.

Motivated by the above observations, we analyze different elements of YOLOv5 architecture to improve detection performance in specific tasks. To the best of our knowledge, there is no work

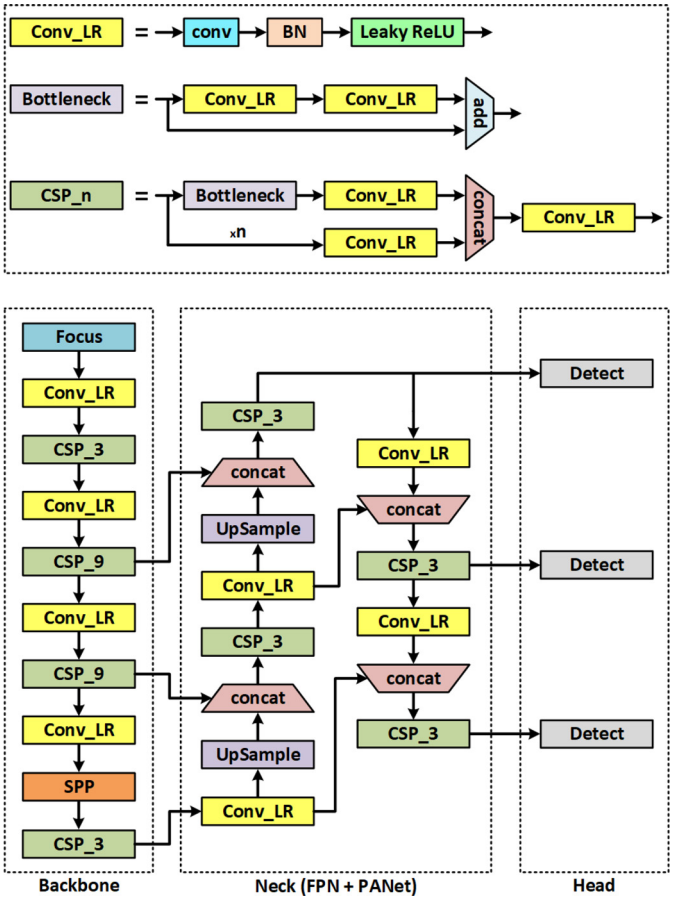

Fig. 1. Original YOLOv5 architecture.

that optimizes the structure of YOLOv5 for autonomous driving by proposing architectural changes for accurate detection of small objects at faster inference time without compromising the detection accuracy of large objects, and with a minimal increase in model complexity.

3.2. YOLOv5

As with the most recent one-stage detection model, YOLOv5 has a strong ability for feature extraction with high detection speed and accuracy. The YOLOv5 series provides four model scales: YOLOv5 - S, YOLOv5 - M, YOLOv5 - L, and YOLOv5 - X, where S is small, M is medium, L is large, and X is xlarge [2]. The network structure of these models is constant, but the modules and convolution kernels are scaled, which alters the complexity and size of each model. In this letter, we analyze and focus on the small variant of YOLOv5 model. The basic architecture of YOLOv5 is illustrated in Fig. 1, including input, backbone, neck, and head.

In the input, YOLOv5 uses Mosaic data augmentation to enhance data for small-scale detection. The first part of the architecture is the backbone network, which consists of the Focus layer [9], BottleNeckCSP [18], and SPP module [35]. The primary role of the Focus layer is to perform image slicing. This reduces information loss during downsampling, simplifies numerical calculations, and boosts training speed. The BottleNeckCSP not only reduces the overall computational burden but also extracts the in-depth information from the features more effectively. The SPP module is used to increase the receptive field of the network by transforming the variable size feature map into a feature vector of fixed size. The second component is the neck network, which applies

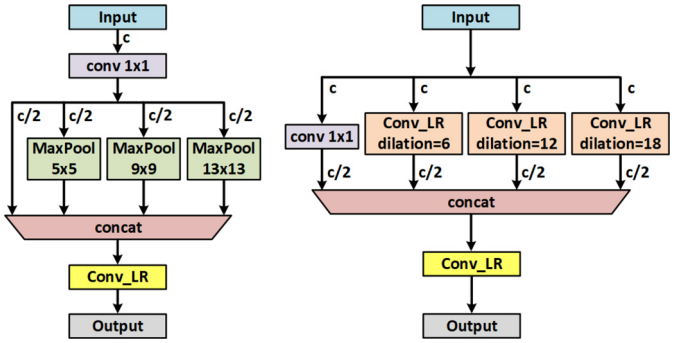

Fig. 2. (a) SPP module (b) Improved SPP module.

PANet [36] and FPN [37] operations. The PANet structure is used for the dense positioning of high-level features. At the same time, FPN provides powerful low-level semantic features by upsampling. Then, various multi-scale features are fused to strengthen the detection ability. The last element is the head network, which performs the final detection of different scales. It outputs a vector containing class probabilities, confidence scores, prediction angle, and bounding box information using anchor boxes.

The detection performance of YOLOv5 is very good, but we observe some major limitations. First, it is primarily aimed at the COCO dataset for generic object detection tasks and does not necessarily apply to specific tasks and related datasets discussed in this work. Second, PANet structure focuses less on information flow between non-adjacent layers, due to which the information is constantly reduced in each aggregation process. Third, the max-pooling operations in the SPP module result in significant information loss, and thus cannot obtain local and global information for localization. Fourth, the information paths that connect the different components in the YOLOv5 architecture limit computational efficiency and are not optimal for extracting relevant features for small-scale objects.

3.3. Proposed architectural changes

The original YOLOv5 model needs to be improved for small object detection in autonomous driving. We introduce several modifications to further enhance detection speed and accuracy without significantly increasing the model complexity.

3.3.1. Improved SPP module

As the depth of the CNN increase, the size of the receptive field becomes larger. Due to the limited size of input images, the feature extraction is repetitive on the large receptive fields [34,38]. Therefore, the SPP module is used to add the corresponding modules to eliminate this problem by fusing the feature maps of different receptive fields. This module combines the local and global features to maximize the expressive ability of feature maps, extends the receiving field of the backbone network, and separates the most important context features for size target detection. For integrating the characteristics of receptive fields of different scales, the SPP uses several max-pooling operations in parallel. This method has shown benefits for improving overall detection accuracy. However, the max-pooling operations fail to capture the spatial information and incur information loss, which results in the inability to accurately locate targets, especially small objects.

To address this issue, we propose an improved SPP module by replacing the pooling function with dilated convolution [39], as shown in Fig. 2. Although both of these operations expand the network receptive field, the pooling reduces the spatial resolution, resulting in the loss of characteristic information. In contrast, the

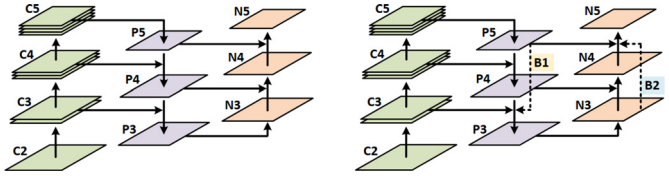

Fig. 3. (a) PANet structure (b) Improved PANet structure.

dilated convolution with different dilation rates enriches the extracted features by capturing multi-scale information required for detecting small targets without losing resolution. The output features are then combined on the same level to enhance feature representation. Accordingly, this module improves the learning ability of the network to accurately locate targets, particularly small objects, while maintaining fast detection speed at a minimal increase in computational cost.

3.3.2. Improved PANet structure

The aim of the neck network is to aggregate the features obtained by the backbone for improving the accuracy of the latter prediction. This structure plays a crucial role in preventing the loss of small object information due to higher abstraction levels. To achieve this, the feature maps are upsampled once again in order to perform aggregation with backbone layers, and reassert influence over the detection [36,40]. The FPN provides deep semantic information to the shallow layers for enhancing detection ability but ignores the location information during the aggregation of shallow and deep features. Whereas PANet structure provides the fusion of both low-level and high-level information. However, the integration method relies on the aggregation of adjacent features, and less attention is given to the exchange of information among non-adjacent layers. Due to this, the spatial information in the non-adjacent layers is continuously reduced with each aggregation.

Therefore, to address the above limitations, we design an improved PANet structure based on the original PANet structure, as depicted in Fig. 3. We add two cross-layer connections (B1 and B2), one in the top-down path of FPN and the other in the bottom-up path of PANet, for integrating non-adjacent and multi-level features. During aggregation, this will allow for more effective use of semantic and shallow location information, enhancing important features of small objects without increasing computational complexity.

3.3.3. Improved information paths

The architectural design of YOLOv5 is simple but needs optimization for computational efficiency and real-time applicability due to its internal arrangement of components. Therefore, we redirect certain connections in order to focus on detecting multi-resolution feature maps. As the input is passed layer-by-layer through the convolutions, the feature maps are extracted. The feature maps generated by the former convolutional layers capture small-scale objects, whereas the feature maps generated by the latter ones capture large-scale objects. The BottleNeckCSP is the most basic block of YOLOv5 and extracts most contextual features, but its current attributes are inefficient for the extraction of deep features, resulting in poor small object detection. Another important aspect is the selection of appropriate activation function, which can limit performance even when adding multiple convolution and normalization layers. Moreover, the head lacks the ability to extract enough shallow features for localizing small objects.

In response to the above problems, we propose an improved Scaled YOLOv5 (iS-YOLOv5), a robust and efficient architecture, the detailed structure of which is described in Fig. 5. For justifying the limitations of the BottleNeckCSP, a new functional block called N-CSP, is introduced by modifying the information paths. We reduce the number of N-CSP blocks in the backbone to adjust network parameters and increase computational speed. Furthermore, we implement Hard Swish activation instead of Leaky ReLU in specific layers of the network. We apply multiple activations to avoid information loss and reduce computational cost according to the input size [41]. On the detection side, we add a detection head for small-scale objects obtained from high-resolution feature maps. In the neck, we optimally adjust the N-CSP blocks to focus on detecting multi-scale features. This will improve the overall detection ability for different scale objects, particularly small targets. Note that our collective modifications barely change the computational complexity while significantly improving detection performance as well as ensuring real-time requirements.

4. Experimental results

In this section, we describe the autonomous driving datasets, training environment, and performance evaluation indicators. Thereafter, we verify the superiority of the proposed method through several experiments.

4.1. Dataset description

We choose BDD100K [42] as our primary dataset to validate the performance of the proposed method. Besides, we analyze the detection performance of traffic signs and traffic lights using TT100K [43] and DTLD [44] datasets, respectively. Both of these datasets are highly challenging for traffic object detection tasks. The TT100K and DTLD datasets provide 30K annotated traffic signs and 200K annotated traffic lights, respectively. The BDD100K dataset comprises 100K self-driving images from different environments in diverse weather scenarios at various day and night time. For several categories, we consider objects as large if their occupying area is more than 112×112 pixels, small if it is less than 48×48 pixels, and medium if it lies between the two thresholds. In this evaluation, we mainly focus on small objects like traffic lights and traffic signs. More than 80% of these objects have an area of less than 48×48 pixels.

4.2. Data augmentation

To enhance the quality of the training data, we apply a series of data augmentation techniques. Through this, a model can learn the characteristics of objects in different scales, lightening, and angles, which can improve the model generalization performance on the unseen data. Among several data augmentation methods [3], we adopt image displacement, linear scaling, horizontal flipping, motion blurring, uniform cropping, and noise adding. In addition, we use Mosaic data augmentation [18], which allows us to train four images instead of one image. The benefit is that it enables training over a single GPU. Fig. 4 depicts the working process of Mosaic data augmentation. To provide a fair comparison of the experimental results, we apply data augmentation techniques to all object detection models.

4.3. Training setup

The hardware environment comprises Intel i9-9900K CPU and NVIDIA Quadro RTX 5000 GPU. At the same time, the software environment is PyTorch v1.8.0 on Ubuntu 18.04.5 OS. We use the SGD optimizer for learning and updating the parameters through the training procedure [9]. Few hyperparameters are set as: the learning rate initialized to 0.01, the training batch size to 4, the momentum to 0.948, the weight decay to 0.0001, and the number of

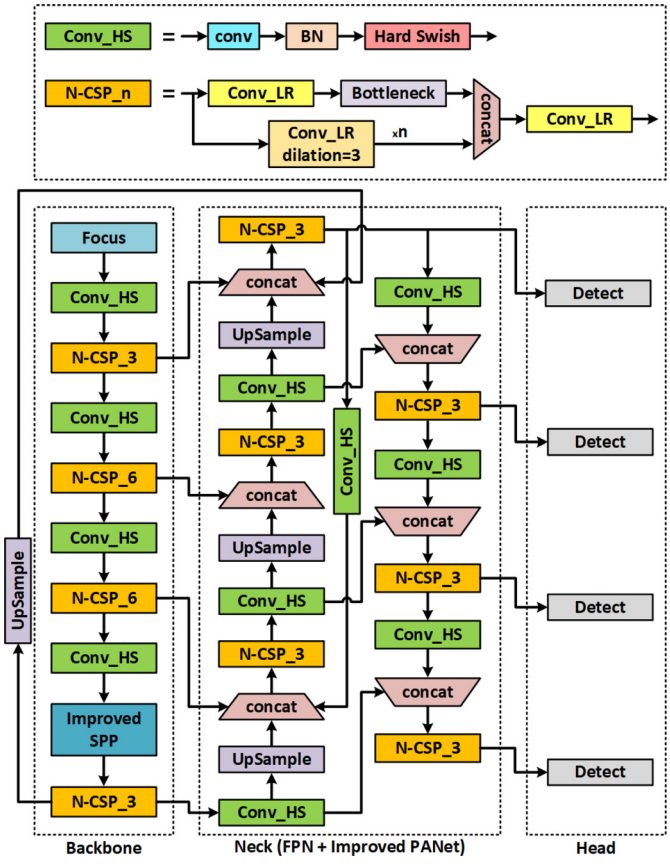

Fig. 4. Implementation of Mosaic data augmentation.

epochs to 100. We leave the remaining configurations as default to the original YOLOv5 model.

4.4. Evaluation metrics

The mean Average Precision (mAP) is the most common evaluation index used to evaluate the detection performance of the network [2,45]. mAP considers both Precision (P) and Recall (R), defined in Eqs. 1 and 2, respectively. AP is the average of precision at different recall values obtained by Precision Recall (PR) curve under a given threshold, defined in Eq. 3. Intersection Over Union (IoU) is the degree of overlap between the predicted box and the actual box, which is set to IoU $\geq$ 0.5. Finally, mAP is the mean of AP over all classes, defined in Eq. 4. In addition, we track computational complexity in terms of Floating Point Operation (FLOPs) and measure inference time per image for validating the model efficiency.

$$Precision\ (P) = \frac{TP}{TP + FP} \tag{1}$$

$$Recall\ (R) = \frac{TP}{TP + FN} \tag{2}$$

Table 1
Results of traffic sign and traffic light detection of the proposed iS-YOLOv5 model.

Model	TT100K		DTLD	
	$AP_{traffic\ sign}$	mAP	$AP_{traffic\ light}$	mAP
YOLOv5 [9]	64.07	67.35	69.16	73.29
iS-YOLOv5	65.93	71.82	72.03	77.14

Table 2
Performance evaluation of the proposed iS-YOLOv5 model.

Model	$AP_{traffic\ sign}$	$AP_{traffic\ light}$	mAP	FPS	GFLOPs
YOLOv5 [9]	50.26	55.37	61.89	56.95	16.47
iS-YOLOv5	53.61	57.08	65.24	59.52	16.83

where TP denotes the number of detections in both the ground truth and the results. FP denotes the number of detections in the results only, not in the ground truth. FN denotes the number of detections in ground truth only, not in results.

$$Average\ Precision\ (AP) = \sum_{k} (R_{k+1} - R_k) \max_{R:\bar{R} \geq R_{k+1}} P(\bar{R}) \tag{3}$$

where $P(\bar{R})$ is the measured P at $\bar{R}$.

$$mean\ Average\ Precision\ (mAP) = \frac{1}{n} \sum_{i=1}^{n} AP_i \tag{4}$$

where n is the number of classes and AP_i is the AP at class i.

4.5. Effectiveness of the proposed model

We present empirical results for the detection of traffic lights and traffic signs in Table 1. In all experiments, mAP is computed for all classes in the dataset. As shown, the proposed iS-YOLOv5 model improves the AP for traffic signs to 65.93% and traffic lights to 72.03% on TT100K and DTLD datasets, respectively. In addition, our model improves the mAP by 4.47% and 3.85% on TT100K and DTLD datasets, respectively.

The performance results of the proposed iS-YOLOv5 model are shown in Table 2. It can be seen from the results that our iS-YOLOv5 increases the detection accuracy for small road objects like traffic signs and traffic lights to 53.61% and 57.08%, respectively. Moreover, the proposed model improves the mAP by 3.35% and inference speed by 2.57 FPS with only 0.36 increase in computations.

We individually show the results for objects of different scales in Table 3. As shown, the proposed iS-YOLOv5 model increases the mAP of large, medium, and small objects by 6.41%, 4.61%, and 1.89%, respectively. It is worth emphasizing that the accuracy difference between large and small objects is quite significant. Therefore, we conclude that the detection accuracy of small objects can be improved without compromising the detection accuracy of medium and large objects.

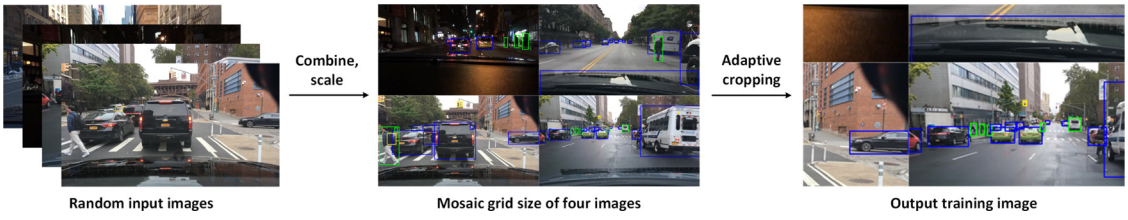

Fig. 5. Detailed structure of the proposed iS-YOLOv5 model.

Table 3
Accuracy of the proposed iS-YOLOv5 model at three different scales.

Model	mAP_{large}	mAP_{medium}	mAP_{small}
YOLOv5 [9]	81.23	68.45	47.26
iS-YOLOv5	87.64	73.06	49.15

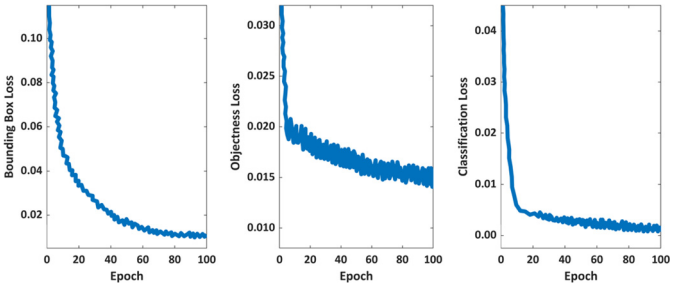

Fig. 6. Validation loss curves for the proposed iS-YOLOv5 model.

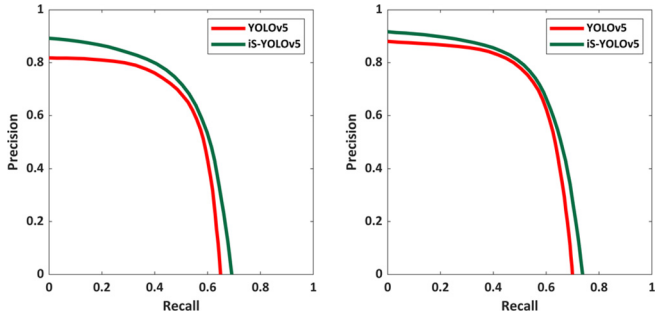

Fig. 7. Comparison of PR curves (a) Traffic Sign (b) Traffic Light.

The validation loss curves are visualized in Fig. 6. In particular, the bounding box, classification, and objectiveness loss curves are obtained during the training process. Furthermore, Fig. 7 presents a comparison in terms of PR curves. As shown, the PR curves of our iS-YOLOv5 have completely enclosed the PR curves of YOLOv5.

We test the applicability of our model in diverse weather scenarios like clear, overcast, partly cloudy, snowy, foggy, and rainy. Table 4 lists the generalization ability of the proposed iS-YOLOv5 model using AP values. It is clear from the results that our model has a strong generalization performance even in very complex environments. This further indicates that our modifications improve performance in complex weather scenarios, which can help extend the current visual perception.

The detection performance of small objects in varying traffic environments is shown in Fig. 8. As observed, when the traffic density increases (top to bottom), the prediction confidence of YOLOv5 decreases and starts to miss targets. In contrast, the proposed iS-YOLOv5 model detects traffic signs and traffic lights with high confidence, even in high traffic scenarios.

Fig. 8. Comparison of detection results (a) YOLOv5 (b) iS-YOLOv5.

4.6. Ablation study of the proposed model

We analyze the contribution of different components in our iS-YOLOv5 model. For this purpose, we conduct ablation experiments and present the results in Table 5. When SPP, PANet, and information paths are applied separately, the detection accuracy increases by 1.83%, 1.05%, and 0.47%, respectively, the inference speed by 0.03 FPS, 0.11 FPS, and 2.43 FPS, respectively, and the computations by 0.32, 0.01, and 0.03, respectively. Note that when our modules are used in combinations, they do not conflict. This clearly shows the impact of the proposed structures that are integrated into our iS-YOLOv5 model.

4.7. Comparison with other detection models

In order to verify the superiority of the proposed iS-YOLOv5 model, we compare our method with several used object detectors. Table 6 compares the accuracy, speed, and complexity of different frameworks under the same settings. Evidently, our iS-YOLOv5 model outperforms the benchmarks by a large margin at a relatively low computational cost. This proves that our model achieves satisfactory results and is thus suitable for real-time detection in autonomous driving applications.

5. Conclusion

In this letter, we study and analyze the effect of different architectural modifications applied to the popular YOLOv5 structure for improving the detection performance of small-scale objects without sacrificing the detection accuracy of large objects. To achieve this, we make refinements for optimizing the flow of information

Table 4
Generalization ability of our iS-YOLOv5 model in different road weather conditions.

Model		Weather scenarios					
		Clear	Partly Cloudy	Overcast	Foggy	Snowy	Rainy
YOLOv5 [9]	$AP_{traffic\ sign}$	54.31	52.44	52.43	51.65	45.29	40.76
	$AP_{traffic\ light}$	57.94	57.51	56.47	53.84	52.63	49.61
iS-YOLOv5	$AP_{traffic\ sign}$	57.82	54.90	54.12	53.27	46.68	41.32
	$AP_{traffic\ light}$	61.73	60.64	58.93	55.76	53.42	50.25

Table 5
Ablation study of different components in our iS-YOLOv5 model.

Improved SPP	Improved PANet	Improved Paths	mAP	FPS	GFLOPs
			61.89	56.95	16.47
✓			63.72	56.98	16.79
	✓		62.94	57.06	16.48
		✓	62.36	59.38	16.50
✓	✓		64.77	57.09	16.80
✓		✓	64.19	59.41	16.82
	✓	✓	63.41	59.49	16.51
✓	✓	✓	65.24	59.52	16.83

Table 6
Performance comparison with state-of-the-art detection models.

Model	mAP	FPS	GFLOPs
SSD [46]	52.18	54.93	95.67
R-FCN [47]	53.02	36.84	294.51
CenterNet [40]	53.69	41.21	63.78
Faster R-CNN [45]	54.84	27.36	234.05
RetinaNet [37]	55.36	24.18	276.73
Cascade R-CNN [38]	57.24	18.45	292.16
YOLOv4 [18]	60.17	42.36	127.85
YOLOv5 [9]	61.89	56.95	16.47
iS-YOLOv5 (ours)	65.24	59.52	16.83

through different network layers. Accordingly, we propose the iS-YOLOv5 model, which is capable of boosting the detection accuracy and speed without greatly increasing the model complexity. We validate the superiority of our iS-YOLOv5 model through extensive experimentation on challenging datasets. In addition, we test the generalization applicability of the proposed model in complex road weather conditions. Using these insights, the current driving systems can be updated to detect small targets like traffic signs and traffic lights in situations where such models are incapable of detecting anything at all. Through this, the detection and perception robustness of an autonomous vehicle can be further extended, resulting in effective planning and decision-making.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] S.S.A. Zaidi, M.S. Ansari, A. Aslam, N. Kanwal, M. Asghar, B. Lee, A survey of modern deep learning based object detection models, Digit. Signal Process. (2022) 103514.
[2] B. Mahaur, N. Singh, K. Mishra, Road object detection: a comparative study of deep learning-based algorithms, Multimed. Tools Appl. 81 (10) (2022) 14247–14282.
[3] K. Tong, Y. Wu, F. Zhou, Recent advances in small object detection based on deep learning: a review, Image Vis. Comput. 97 (2020) 103910.
[4] G. Cheng, Y. Yao, S. Li, K. Li, X. Xie, J. Wang, X. Yao, J. Han, Dual-aligned oriented detector, IEEE Trans. Geosci. Remote Sens. 60 (2022) 1–11.
[5] F. Xiaolin, H. Fan, Y. Ming, Z. Tongxin, B. Ran, Z. Zenghui, G. Zhiyuan, Small object detection in remote sensing images based on super-resolution, Pattern Recognit. Lett. 153 (2022) 107–112.
[6] G. Cheng, X. Yuan, X. Yao, K. Yan, Q. Zeng, J. Han, Towards large-scale small object detection: survey and benchmarks (2022) arXiv preprint arXiv:2207.14096.
[7] B. Mahaur, K. Mishra, N. Singh, Improved residual network based on norm-preservation for visual recognition, Neural Netw. 157 (2023) 305–322.
[8] I. Rio-Torto, K. Fernandes, L.F. Teixeira, Understanding the decisions of cnns: an in-model approach, Pattern Recognit. Lett. 133 (2020) 373–380.
[9] G. Jocher, et al., ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO Export and Inference, 2022, doi:10.5281/zenodo.6222936.
[10] P. Jiang, D. Ergu, F. Liu, Y. Cai, B. Ma, A review of yolo algorithm developments, Procedia Comput. Sci. 199 (2022) 1066–1073.
[11] X. Han, J. Chang, K. Wang, Real-time object detection based on yolo-v2 for tiny vehicle object, Procedia Comput. Sci. 183 (2021) 61–72.
[12] B. Liu, F. He, S. Du, J. Li, W. Liu, An advanced yolov3 method for small object detection (2022) arXiv preprint arXiv:2212.02809.
[13] G. Li, W. Fan, H. Xie, X. Qu, Detection of road objects based on camera sensors for autonomous driving in various traffic situations, IEEE Sens. J. 22 (24) (2022) 24253–24263.
[14] R. Wang, Z. Wang, Z. Xu, C. Wang, Q. Li, Y. Zhang, H. Li, A real-time object detector for autonomous vehicles based on yolov4, Comput. Intell. Neurosci. 2021 (2021).
[15] Y. Li, et al., A deep learning-based hybrid framework for object detection and recognition in autonomous driving, IEEE Access 8 (2020) 194228–194239.
[16] Y. Cai, et al., Yolov4-5d: an effective and efficient object detector for autonomous driving, IEEE Trans. Instrum. Meas. 70 (2021) 1–13.
[17] S. Du, P. Zhang, B. Zhang, H. Xu, Weak and occluded vehicle detection in complex infrared environment based on improved yolov4, IEEE Access 9 (2021) 25671–25680.
[18] A. Bochkovskiy, C.-Y. Wang, H.-Y.M. Liao, Yolov4: optimal speed and accuracy of object detection (2020) arXiv preprint arXiv:2004.10934.
[19] L. Gou, et al., Vatld: a visual analytics system to assess, understand and improve traffic light detection, IEEE Trans. Vis. Comput. Graph. 27 (2) (2020) 261–271.
[20] H. Zhang, L. Qin, J. Li, Y. Guo, Y. Zhou, J. Zhang, Z. Xu, Real-time detection method for small traffic signs based on yolov3, IEEE Access 8 (2020) 64145–64156.
[21] W. Omar, I. Lee, G. Lee, K. Park, Detection and localization of traffic lights using yolov3 and stereo vision, ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 43 (2020) 1247–1252.
[22] J. Lian, Y. Yin, L. Li, Z. Wang, Y. Zhou, Small object detection in traffic scenes based on attention feature fusion, Sensors 21 (9) (2021) 3031.
[23] Z.-Z. Wang, K. Xie, X.-Y. Zhang, H.-Q. Chen, C. Wen, J.-B. He, Small-object detection based on yolo and dense block via image super-resolution, IEEE Access 9 (2021) 56416–56429.
[24] A. Benjumea, I. Teeti, F. Cuzzolin, A. Bradley, Yolo-z: improving small object detection in yolov5 for autonomous vehicles (2021) arXiv preprint arXiv:2112.11798.
[25] W. Liu, Z. Wang, B. Zhou, S. Yang, Z. Gong, Real-time signal light detection based on yolov5, in: IOP Conference Series: Earth and Environmental Science, volume 769, IOP Publishing, 2021, p. 042069.
[26] J. Chen, K. Jia, W. Chen, Z. Lv, R. Zhang, A real-time and high-precision method for small traffic-signs recognition, Neural Comput. Appl. 34 (3) (2022) 2233–2245.
[27] Q. Su, H. Wang, M. Xie, Y. Song, S. Ma, B. Li, Y. Yang, L. Wang, Real-time traffic cone detection for autonomous driving based on yolov4, IET Intel. Transport Syst. (2022).
[28] I. Katsamenis, et al., Tracon: a novel dataset for real-time traffic cones detection using deep learning (2022) arXiv preprint arXiv:2205.11830.
[29] Z. Jiang, L. Zhao, S. Li, Y. Jia, Real-time object detection method based on improved yolov4-tiny (2020) arXiv preprint arXiv:2011.04244.
[30] S. Li, Y. Li, Y. Li, M. Li, X. Xu, Yolo-firi: improved yolov5 for infrared image object detection, IEEE Access 9 (2021) 141861–141875.
[31] T. Liang, et al., Alodad: an anchor-free lightweight object detector for autonomous driving, IEEE Access 10 (2022) 40701–40714.
[32] J. Ning, J. Wang, Automatic driving scene target detection algorithm based on improved yolov5 network, in: 2022 International Conference on Computer Network, Electronic and Automation (ICCNEA), IEEE, 2022, pp. 218–222.
[33] Z. Ge, S. Liu, F. Wang, Z. Li, J. Sun, Yolox: exceeding yolo series in 2021 (2021) arXiv preprint arXiv:2107.08430.
[34] X. Wang, X. Hu, C. Chen, S. Peng, Regularizing deep networks with label geometry for accurate object localization on small training datasets, Pattern Recognit. Lett. 154 (2022) 53–59.
[35] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1904–1916.

[36] T.-Y. Lin, et al., Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.
[37] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.
[38] Z. Cai, N. Vasconcelos, Cascade r-cnn: delving into high quality object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 6154–6162.
[39] X. Lin, C.-T. Li, V. Sanchez, C. Maple, On the detection-to-track association for online multi-object tracking, Pattern Recognit. Lett. 146 (2021) 200–207.
[40] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, Q. Tian, Centernet: keypoint triplets for object detection, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 6569–6578.
[41] P. Singh, M. Varshney, V. Namboodiri, Cooperative initialization based deep neural network training, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2020, pp. 1141–1150.
[42] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, T. Darrell, Bdd100k: a diverse driving dataset for heterogeneous multitask learning, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 2636–2645.
[43] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, S. Hu, Traffic-sign detection and classification in the wild, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2110–2118.
[44] A. Fregin, J. Muller, U. Krebel, K. Dietmayer, The driveu traffic light dataset: introduction and comparison with existing datasets, in: 2018 IEEE international conference on robotics and automation (ICRA), IEEE, 2018, pp. 3376–3383.
[45] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst. 28 (2015).
[46] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, Ssd: single shot multibox detector, in: European conference on computer vision, Springer, 2016, pp. 21–37.
[47] J. Dai, Y. Li, K. He, J. Sun, R-Fcn: object detection via region-based fully convolutional networks, Adv. Neural Inf. Process. Syst. 29 (2016).