

TOPICAL REVIEW

A Review of YOLO Algorithm and Its Applications in Autonomous Driving Object Detection

JIAPEI WEI^{1,2}, AZIZAN AS'ARRY², KHAIRIL ANAS MD REZALI²,
MOHD ZUHRI MOHAMED YUSOFF³, HAOHAO MA², (Graduate Student Member, IEEE),
AND KUNLUN ZHANG²

¹Department of Automotive Engineering, Tianshui Normal University, Tianshui 741001, China

²Department of Mechanical and Manufacturing Engineering, Universiti Putra Malaysia, Serdang 43400, Malaysia

³Laboratory of Biocomposite Technology, Institute of Tropical Forestry and Forest Products, Universiti Putra Malaysia, Serdang 43400, Malaysia

Corresponding author: Azizan As'arry (zizan@upm.edu.my)

This work was supported in part by the Universiti Putra Malaysia (UPM) and the Malaysian Ministry of Higher Education (MOHE); in part by the Geran Putra Inisiatif (GPI) Fund under Grant GPI/2024/9794100; and in part by the Innovation Fund Project of Gansu Provincial Education Department, China, under Project 2023B-140.

ABSTRACT Object detection in autonomous driving scenarios represents a significant research direction within artificial intelligence. Real-time and accurate object detection and recognition are crucial in ensuring autonomous vehicles' safe and stable operation. In recent years, the continuous introduction of the YOLO series of algorithms and their enhanced models has led to remarkable performance in autonomous driving object detection. From YOLOv1 to YOLOv12, detection accuracy has improved significantly, with mAP increasing from approximately 63.4% to over 80% and inference speed exceeding 100 FPS in lightweight versions such as YOLOv8n and YOLOv10. This paper reviews the YOLO algorithm and its application in object detection in autonomous driving scenarios. Firstly, the development and distinctions among the YOLO series of detection algorithms are explained, and their performance is analyzed. Secondly, the strategies for improving YOLO-based models across the input, feature extraction, and prediction stages are summarized. Thirdly, the research status and application of the YOLO algorithm in autonomous driving object detection are elaborated upon from the perspectives of traffic vehicles, pedestrians, traffic signs, traffic lights, and lane lines, with comparisons and analyses of performance metrics such as accuracy and real-time performance. Finally, considering the current challenges in autonomous driving object detection, the development trajectory and prospects of the YOLO algorithm are summarized and discussed.

INDEX TERMS Autonomous driving, object detection, YOLO algorithm, applications.

I. INTRODUCTION

Autonomous driving uses sensors such as cameras and radars to sense the surrounding environment. It can quickly and accurately detect and judge various objects so that the vehicle can autonomously control its behavior and achieve autonomous driving [1], [2]. Autonomous driving, due to its intelligence, can reduce human errors to a large extent and achieve energy conservation and emission reduction [3]. Waymo, a US research institute, analyzed the data of autonomous driving vehicles and found that compared with

human driving, the accident rate of “any injury report” of autonomous driving vehicles was reduced by 80%, and the accident rate of “police report” was reduced by 55% [4].

Accurately understanding the surrounding environment is crucial to ensuring the safety and reliability of autonomous driving. Among the many environmental perception tasks, object detection is one of the key components. An integral part of computer vision systems, object detection can automatically find and mark things in pictures or video frames [5], [6], [7]. The main task of autonomous driving object detection is to identify, locate, and track various types of objects in traffic scenes (such as vehicles, pedestrians, and traffic signals), and provide real-time and accurate information such

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaosong Hu².

as their category, location, motion status, to provide a basis for autonomous driving decisions [8], [9], [10]. (Note: In this paper, “real-time” refers to achieving an inference speed of at least 30 frames per second under acceptable accuracy trade-offs, which is considered sufficient to meet the real-time perception requirements in autonomous driving scenarios.)

With the rapid advancement of deep learning, algorithm-based object detection has become a research hotspot in this field. Fig. 1 shows the number of publications related to object detection in autonomous driving, reflecting the growing research interest in this field.

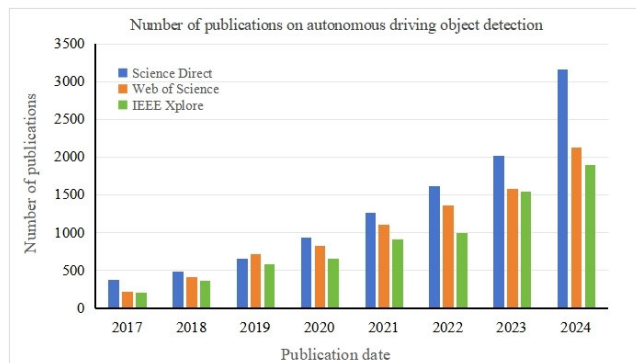


FIGURE 1. The annual trend of research publications on object detection in autonomous driving from 2017 to 2024 was collected using keyword queries such as “autonomous driving object detection” from databases including Science Direct, Web of Science, and IEEE Xplore. The increasing number reflects growing academic attention in this area.

In recent years, numerous scholars have conducted reviews on object detection in autonomous driving. Pravallika et al. [11] explored the application of 3D (three-dimensional) object detection techniques [12], [13] in autonomous driving, focusing on the advantages and limitations of deep learning frameworks in 3D detection. Preeti and Rana [14] reviewed AI-based object detection techniques for autonomous driving, summarizing various deep learning frameworks and discussing the significance of sensor fusion, real-time processing, and explainable AI in this field. Gheorghe et al. [15] analyzed the evolution of the YOLO (you only look once) algorithm [16] and its applications across different domains, focusing on road traffic, autonomous driving, and industrial manufacturing. Diwan et al. [17] provided a comprehensive review of YOLO’s development in object detection, covering its challenges, architectural improvements, commonly used datasets, and application scenarios. Ruan et al. [18] reviewed the detection methods of occluded objects (vehicles, pedestrians, and traffic signs) in complex scenes in autonomous driving, compared the advantages and disadvantages of different detection technologies, and pointed out the direction of future development. Flores-Calero et al. [19] systematically reviewed YOLO’s applications in traffic sign detection and recognition, analyzing its technical principles, architectural evolution, and performance across various traffic scenarios. They further explored the advantages and challenges of YOLO in complex environments.

Existing review papers on object detection in autonomous driving typically focus either on algorithmic advancements or application-specific implementations, with few providing a comprehensive analysis integrating both aspects. This paper addresses this gap by reviewing the latest developments and improvement strategies of the YOLO series while systematically analyzing its diverse applications in autonomous driving. Additionally, it explores current challenges and future research directions. The key contributions of this study are as follows:

- 1) It comprehensively analyzes the innovations and differences across YOLO versions, from YOLOv1 to YOLOv12.
- 2) It compares eight common improvement strategies in the input, feature extraction, and prediction stages of the YOLO model, evaluating their advantages and limitations.
- 3) It examines the application of YOLO in autonomous driving object detection, covering the identification of traffic vehicles, pedestrians, traffic signs, traffic lights, lane lines, and multi-sensor fusion detection, enhancing the latest review studies in this field.
- 4) It highlights the challenges the YOLO algorithm faces in autonomous driving. It proposes development directions such as improving the ability to detect small objects, introducing multi-scale object detection, exploring multi-modal information fusion, lightweight structure design, developing diverse datasets, and providing researchers with research ideas and directions.

II. OVERVIEW OF OBJECT DETECTION

A. OBJECT DETECTION ALGORITHMS

Object detection has evolved through two major phases: traditional methods and deep learning-based approaches [17], [20].

Traditional object detection is mainly based on manual features and machine learning algorithms, such as VJ (viola-jones) [21], [22], HOG (histogram of oriented gradient) [23], [24], DPM (deformable parts model) [25], [26]. While these approaches achieved moderate success, they were limited by feature extraction complexity and computational inefficiency.

Since introducing R-CNN (region-based convolutional neural network) [27] in 2014, deep learning-based methods have become the dominant approach, categorized into two-stage and one-stage detection frameworks [28].

Two-stage detection first generates region proposals to identify areas containing objects and then classifies them using a deep network. This coarse-to-fine approach enhances accuracy but increases computational complexity. It emphasizes selective region proposal strategies to refine object detection. Representative models include R-CNN, SPP-Net (spatial pyramid pooling networks) [29], Fast R-CNN [30], Faster R-CNN [31], R-FCN (region-based fully convolutional network) [32], and Mask R-CNN [33].

In 2016, the one-stage detection paradigm, exemplified by YOLOv1, emerged. Unlike two-stage methods, one-stage detectors classify and localize objects in a single forward

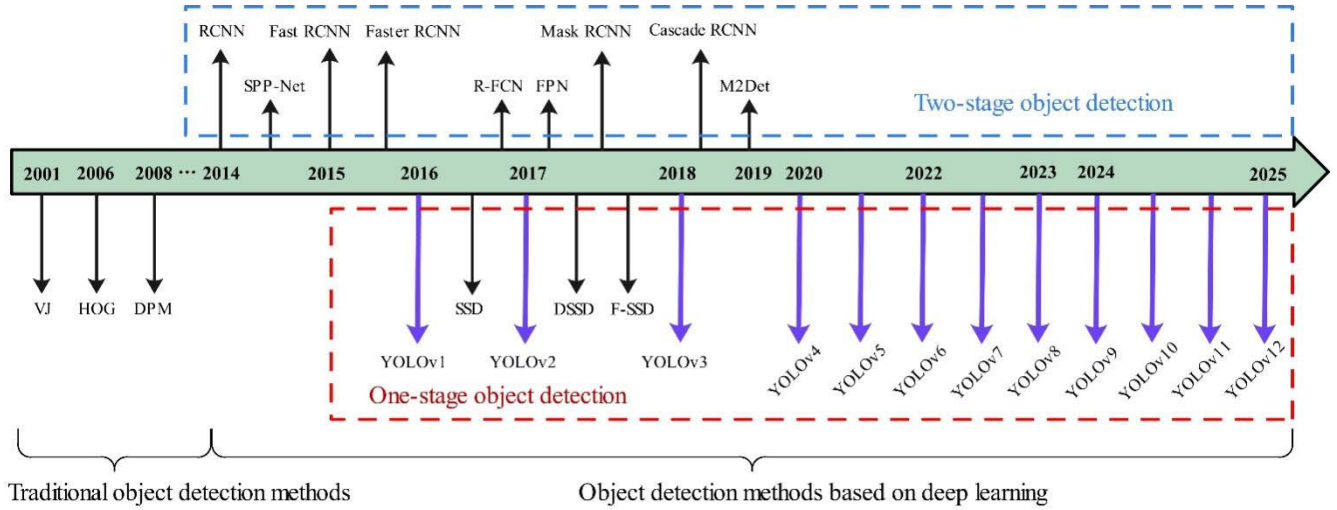


FIGURE 2. The evolution timeline of object detection algorithms illustrates the transition from traditional methods to deep learning-based frameworks, highlighting key innovations and representative models introduced from 2001 (VJ) to 2025 (YOLOv12).

pass across all spatial regions. This end-to-end approach significantly reduces computational overhead, improving detection speed while maintaining competitive accuracy. Notable one-stage models include the YOLO series [34], [35], SSD (single shot multiBox detector) [36], DSSD (deformable single shot detector) [37], RetinaNet [38], DETR (detection transformer) [39] and Sparse R-CNN [40]. Fig. 2 shows the evolution of some mainstream object detection algorithms.

B. OBJECT DETECTION DATASETS AND EVALUATION METRICS

1) OBJECT DETECTION DATASETS

Large-scale datasets are a prerequisite for the success of deep learning and can measure the performance of different algorithms [41], [42]. Commonly used datasets for object detection include PASCAL VOC [43], [44], MS-COCO [45], [46], ImageNet [47], KITTI [48], [49], Open Images [50], nuScenes [51], among others.

In recent years, several newly released and more comprehensive datasets have emerged to support advanced research in autonomous driving. For example, the nuScenes [51] 2024 update offers a large-scale, multi-modal dataset that includes high-quality LiDAR (light detection and ranging), radar, and camera data under diverse weather, lighting, and traffic conditions, along with improved annotations and extended urban driving scenarios [52], [53], [54]. Likewise, the Waymo Open Dataset [55], [56] provides high-resolution, long-range LiDAR sequences and detailed HD (high definition) map context, supporting robust training and benchmarking in real-world scenarios. Moreover, datasets such as Argoverse 2 [57], [58] and PandaSet [59] incorporate map priors, semantic segmentation, and full-stack sensor suites, enriching the complexity and realism of perception challenges. Incorporating such datasets allows for a more

rigorous evaluation of YOLO-based models regarding generalization ability, robustness, and applicability in real-world autonomous driving environments. Table 1 summarizes standard datasets used in autonomous driving object detection and their characteristics.

2) EVALUATION METRICS

Object detection has well-established evaluation metrics. We typically assess classification accuracy using accuracy (A), precision (P), recall (R), average precision (AP), and mean average precision (mAP) [65]. We commonly measure localization accuracy by the intersection over union (IoU), and we evaluate processing speed using frames per second (FPS) [66].

We categorize detection results into positive and negative samples to define evaluation metrics. A true positive (TP) occurs when we correctly identify a positive sample, while a false negative (FN) occurs when we misclassify a positive sample as negative. Similarly, a true negative (TN) refers to a correctly identified negative sample, whereas a false positive (FP) occurs when a negative sample is mistakenly classified as positive [67]. The definitions of the key evaluation metrics are as follows.

1) Accuracy refers to the proportion of correctly predicted samples among all predictions. The calculation formula is as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

2) Precision refers to the proportion of correctly predicted positive samples among all predicted positive samples. The calculation formula is as follows:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

TABLE 1. Common datasets for object detection in autonomous driving.

Dataset	Introduction	Commonly detected objects	Source
KITTI [48], [49]	Applicable to object detection, tracking, and semantic segmentation.	Vehicles, pedestrians	Karlsruhe Institute of Technology and Toyota Technological Institute
nuScenes [51]	A dataset includes images, LiDAR, and radar data with 3D information.	Vehicles, pedestrians, traffic signs	Aptiv Corporation
BDD100K [60]	It covers various weather and lighting conditions and is currently the most significant autonomous driving scenario dataset.	Vehicles, pedestrians, traffic signs	University of California, Berkeley
MS-COCO [45], [46]	Annotated for object detection, segmentation, and key points.	Vehicles, pedestrians	Microsoft Research
PASCAL VOC [43], [44]	A classic computer vision dataset used for object detection, classification, and segmentation tasks.	Vehicles, pedestrians	University of Oxford
Waymo Open [55], [56]	A large-scale autonomous driving dataset containing three different road scenarios.	Vehicles, pedestrians, traffic signs	Waymo LLC
WiderPerson [61]	It covers various scenarios and occlusion conditions.	Pedestrians	CASIA, China
LISA [62]	Images and videos of 47 types of US traffic signs were collected from different cameras.	Pedestrians, vehicles, lane markings	Karlsruhe Institute of Technology, Germany
Cityscapes [63]	Semantic segmentation dataset for urban street scenes.	Vehicles, pedestrians	Freie Universität Berlin, Germany
ApolloScape [64]	Used for object detection, semantic segmentation, and depth estimation with 3D information.	Vehicles, pedestrians, lane markings	Baidu, China
INRIA [23]	Each image is annotated with bounding box coordinates and size information for pedestrian areas.	Pedestrians	INRIA, French

3) Recall refers to the proportion of correctly predicted positive samples among all actual positive samples. The calculation formula is as follows:

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

4) Average Precision (AP) is calculated as the area under the Precision-Recall (P-R) curve, representing the average accuracy of detection across different recall levels. It reflects the accuracy of a single category. The calculation formula is as follows:

$$AP = \int_0^1 Precision(R) dR \quad (4)$$

5) Mean Average Precision (mAP) is the mean value of AP across all object categories, providing an overall assessment of detection performance. A higher mAP indicates better model performance. The calculation formula is as follows:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (5)$$

where N represents the number of categories, AP_i represents the average precision (AP) of the i -th category.

6) Intersection over Union (IoU) measures the overlap between the predicted bounding box and the ground truth bounding box. It is calculated as the ratio of their intersection to their union, reflecting the accuracy of object localization. A higher IoU indicates a more precise prediction.

$$IoU = \frac{Area\ of\ inter\ section}{Area\ of\ Union} \quad (6)$$

7) Frames Per Second (FPS) represents the number of images processed per second, reflecting the speed performance of the object detection model. A higher FPS value signifies faster detection.

8) The F1 Score is defined as the harmonic mean of precision and recall, providing a balanced measure that combines both metrics. The calculation formula is as follows:

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (7)$$

III. YOLO SERIES MODELS

In 2016, Redmon et al. [16] proposed the YOLOv1 algorithm, which innovatively transformed the object detection task into a regression problem. By leveraging a CNN (convolutional neural network), YOLOv1 directly predicted object classes and bounding boxes, enabling real-time object detection. Subsequently, multiple versions of YOLO have been introduced with continuous improvements [34], [68], [69], [70], [71], [72]. This section overviews YOLO's fundamental idea, evolution, network architecture, and improvement strategies.

A. FUNDAMENTAL IDEA OF YOLO

The YOLO algorithm reframes object detection as a regression task, enabling the prediction of the class and position of all objects within an image via a single network pass [73]. Its fundamental concept involves utilizing the entire image as input and conducting direct inference through a CNN. This approach facilitates real-time object detection in an end-to-end manner. Fig. 3 illustrates the basic framework of the YOLO model.

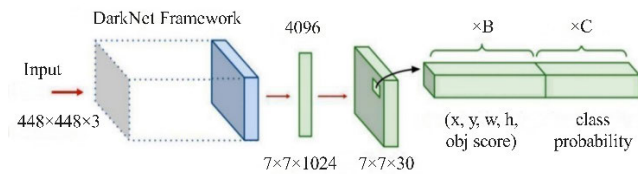


FIGURE 3. The basic framework of the YOLO model.

Initially, the input image is scaled to a resolution of 448×448 and then processed through a CNN for feature extraction. The CNN segments the image into an $S \times S$ grid, with each grid cell tasked with detecting objects whose centers are located within that cell. Each grid cell predicts B bounding boxes and their corresponding confidence values. The confidence value indicates the likelihood of an object being present as well as the precision of the bounding box. Each bounding box prediction consists of five elements: (x, y, w, h, c) , where (x, y) represents the center coordinates, (w, h) denote the width and height, and c is the confidence score. Each grid cell predicts $(B \times 5 + C)$ values, where C is the number of object categories. Finally, non-maximum suppression is applied to eliminate redundant bounding boxes and refine predictions. This fundamental approach has been inherited and further optimized in subsequent versions of the YOLO series.

The YOLO model is structured into three main stages to achieve the above process: input, feature extraction, and prediction [15]. Fig. 4 shows the overall structure of the YOLO model.

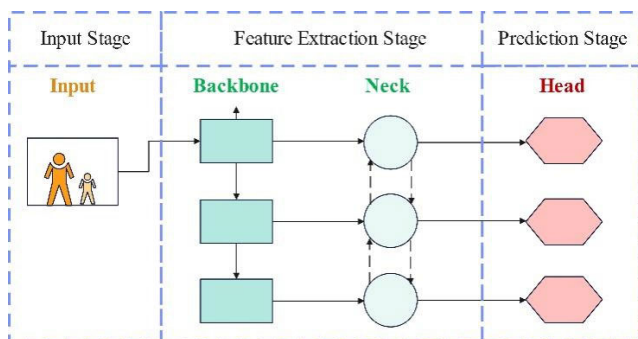


FIGURE 4. The overall structure of the YOLO model.

The input stage accepts data and employs data augmentation and preprocessing techniques to boost the model’s capacity to adapt to diverse input scenarios. Various YOLO models utilize different resolutions for their input data.

The feature extraction stage typically employs a pre-trained CNN as the backbone. The backbone effectively learns and extracts essential features from input images by optimizing the network structure and parameter settings. Starting from YOLOv2, researchers introduced an additional neck layer following the backbone. This layer refines feature maps, improving adaptability to different tasks and datasets. The neck layer was first introduced in YOLOv3 and has since become an integral part of subsequent YOLO models.

The prediction stage, corresponding to the head layer, generates anchor boxes of different sizes based on the extracted feature maps. NMS (non-maximum suppression) is then applied to eliminate redundant bounding boxes, ultimately producing the final object classification results.

B. EVOLUTION OF THE YOLO MODEL

In 2016, Redmon et al. [16] proposed YOLOv1, the inaugural network in the YOLO family. It comprised 24 convolutional layers designed for feature extraction from images and two fully connected layers dedicated to predicting bounding boxes and class probabilities. All layers used the Leaky ReLU (rectified linear unit) activation function except the last layer, which used the linear activation function [74]. During inference, the confidence score for each bounding box was obtained by multiplying the predicted confidence value with the corresponding conditional class probability. Unlike contemporary two-stage detectors such as Faster R-CNN, which first generate region proposals and then perform classification, YOLOv1 achieves significantly higher inference speed. However, it has difficulty detecting small objects and achieving high localization accuracy.

To address these limitations, Redmon et al. [75] proposed YOLOv2 in 2017. It builds a new backbone network darknet-19 based on YOLOv1, adopts multi-scale training, improves the predicted bounding box, and designs the passthrough layer. In addition, introducing the anchor box mechanism, clustering to set the prior box, and predicting the direct coordinate improve the detection accuracy. Compared with the SSD algorithm of the same period, it focuses on real-time object detection and can better handle multi-scale features, although SSD performs well in small object detection.

In 2018, Redmon and Farhadi [76] proposed YOLOv3, which has a balanced speed and accuracy. It adopts the darknet-53 network and introduces residual connections to enhance feature extraction efficiency. In addition, the FPN (feature pyramid network) [77] is used for multi-scale feature fusion so that YOLOv3 can capture objects of different sizes and improve the detection ability of objects of different scales. Compared to RetinaNet, which solves the category imbalance problem by introducing Focal Loss, YOLOv3 retains a simpler single-stage training strategy, giving priority to real-time detection speed without adding additional complexity.

In 2020, Bochkovskiy et al. [78] established YOLOv4 based on the original YOLOv3. Compared to YOLOv3, YOLOv4 introduced data augmentation techniques, regularization methods, complete IoU (intersection over union) loss function [79], [80], self-adversarial training, and cross-minibatch normalization to improve detection accuracy. In addition, they implemented some architectural optimizations, including the Mish [81] activation function, CSP (cross-stage partial) [82] connections, PAN (path aggregation network) [83], and SPP. Meanwhile, detection methods such as Mask R-CNN are dominant but computationally more

expensive. YOLOv4 successfully narrows the accuracy gap with two-stage detectors while maintaining faster inference speed, further consolidating the advantage of the YOLO model in real-time applications.

In the same year, Ultralytics introduced YOLOv5 [84], [85], a lightweight model with enhanced computational speed and reduced memory consumption. It was the first version developed using the Pytorch [86] framework, making it more accessible for training, deployment, and practical applications. YOLOv5 integrated various enhancements: the input stage utilized mosaic data augmentation, adaptive anchor box calculation, and adaptive image scaling. The backbone network integrated Focus [87] and CSP modules, while the neck layer featured CSP structures. In contrast, DSSD attempts to improve the performance of SSD by adding a deconvolution module. It focuses on better detection of small objects but brings significant computational overhead, while YOLOv5 achieves a more effective balance between accuracy, speed, and resource consumption and is more conducive to practical deployment.

To accommodate various deployment scenarios and balance trade-offs between detection accuracy and computational efficiency, YOLOv5 introduced multiple model variants differentiated by size and complexity. These variants are denoted by suffixes: “n” (nano), “s” (small), “m” (medium), “l” (large), and “x” (extra large). Specifically, the “n” models are extremely lightweight, objecting edge devices with limited resources; “s” models offer a balance favoring speed; “m” models aim for an intermediate trade-off; “l” models prioritize higher detection accuracy; and “x” models maximize detection performance with increased computational cost. This naming convention has been inherited and further refined in subsequent YOLO versions such as YOLOv8.

In 2022, Li et al. [88] proposed YOLOv6, which is based on the RepVGG-style [89] structure and a Rep-PAN neck layer for enhanced feature fusion in the feature extraction stage. In the detection stage, a decoupled head was introduced, leading to the development of the efficient decoupled detection head, which improved computational efficiency. Additionally, self-distillation strategies [90] were incorporated into regression and classification tasks, leveraging the SimOTA label assignment strategy and the SIOU loss function [91] to reduce regression complexity, accelerate network convergence, and enhance regression accuracy.

In the same year, to enhance the deployment of YOLO on mobile GPU devices, Wang and Alexey et al. [92] introduced YOLOv7. This version introduced data augmentation methods such as Mixup [93], Copy-Paste [94], and Paste-In. In the feature extraction stage, the backbone was redesigned with BConv, E-ELAN, and MPConv layers to improve feature representation. Compared with contemporary methods like DETR, which introduced transformer-based end-to-end detection pipelines but suffered from long convergence times

and slower inference speeds, YOLOv7 maintained excellent detection speed and competitive accuracy, making it more suitable for high-frequency perception tasks.

In 2023, the YOLOv5 team introduced the YOLOv8 [95]. YOLOv8 used the improved CSPDarknet53 [95] as the backbone network and introduced the CSP structure. Dividing the feature map into two parts for alternating processing achieves faster reasoning speed and higher feature extraction capabilities. Meanwhile, YOLOv8 employs PANet (path aggregation network) [96] as the neck network to merge feature maps from various scales, thereby improving the model's ability to detect objects across multiple scales. In addition, the head part changes from the original coupling head to the decoupling head and from the anchor-based YOLOv5 to the anchor-free. This change makes the model no longer dependent on the design of the prior frame, thereby simplifying the training process and improving the model's generalization ability.

In 2024, Wang et al. [97] proposed an efficient object detection algorithm, YOLOv9. Its backbone network used efficientRepV2 and introduces the concept of PGI (progressive gradient interpolation). At the same time, a lightweight network architecture, GELAN (generalized efficient layer aggregation network) [98], is designed to optimize network parameters' utilization and computational efficiency based on gradient path planning.

In the same year, Tsinghua University in China developed the YOLOv10 model [99]. This version uses EfficientRepV3 as the backbone network, which is more lightweight and introduces a more profound hierarchical feature extraction method. Meanwhile, approaches like Sparse R-CNN achieved impressive detection accuracy in complex environments but lacked the real-time inference capabilities that YOLO models consistently prioritized.

In September 2024, Ultralytics released the YOLOv11 model [100]. The main feature is that the backbone network replaces the previous C2f module with C3k2, significantly improving computational efficiency and enabling YOLOv11 to extract features more quickly when processing images. In addition, a C2PSA module is added after the SPPF module. This design helps to enhance the feature extraction capability and improves the model's performance [101], [102].

In February 2025, researchers [103] from the University at Buffalo and the University of Chinese Academy of Sciences proposed the YOLOv12 model, the latest version of the YOLO series. YOLOv12 introduces the Area Attention module, which reduces the complexity of attention calculation by dividing the feature map into vertical or horizontal areas while maintaining a large receptive field. At the same time, the residual efficient layer aggregation network is proposed to solve the optimization challenges introduced by the attention mechanism and enhance the feature aggregation capability [104]. In addition, the traditional attention architecture is optimized, including the introduction of Flash attention [105], [106], [107] to improve memory access

TABLE 2. Architectural Evolution, Key Improvements, Benefits, and Evolutionary Notes of YOLO Versions from YOLOv1 to YOLOv12.

Version	Backbone network	Neck network	Head network	Anchor box	Key improvements	Benefits	Evolution notes
YOLOv1 [16]	GoogleNet-based feature extraction	-	Fully connected network	Bounding box	Unified detection, direct regression	Real-time detection but limited small object accuracy	First direct regression for object detection
YOLOv2 [75]	Darknet-19 backbone with BN and ReLU activation	-	Coupled head	Anchor-based	Anchor boxes, Batch Normalization	Improved localization accuracy and robustness with anchor boxes	Introduced anchor-based prediction
YOLOv3 [76]	Darknet-53 backbone with BN and Leaky ReLU	FPN for multi-scale feature integration	Coupled head	Anchor-based	Multi-scale detection, residual connections	Better performance for small objects; multi-scale detection	Adopted FPN for multi-scale
YOLOv4 [108]	CSPDarknet53 backbone with Mish	PANet and SPP for multi-scale enhancement	Coupled head	Anchor-based	Mosaic augmentation, self-adversarial training	Balanced accuracy and speed, robust training techniques	CSP enhanced backbone
YOLOv5 [109]	CSPDarknet53 with Focus module to reduce computation	C3 modules for enriched residual connections	Coupled head	Anchor-based	Auto anchor, Focus module, lightweight models	Lightweight and flexible for deployment on edge devices	Flexible multi-scale models (n/s/m/l/x)
YOLOv6 [110]	EfficientRep backbone for faster training and inference	Rep-PAN neck for speed optimization	Decoupled head	Anchor-free	Anchor-free design, SimOTA label assignment	Real-time performance enhanced	Optimization for real-time systems
YOLOv7 [92]	ELAN backbone for controlled gradient flow	SPP + E-ELAN for feature fusion	Decoupled head	Anchor-based	Extended ELAN architecture, bag-of-freebies	High accuracy and speed with efficient design	Extended efficient layer aggregation
YOLOv8 [111], [112]	CSPDarknet53 with C2f module for improved gradient flow	BiFPN + PAN for feature integration	Decoupled head	Anchor-free	Anchor-free structure, enhanced backbone	Higher mAP and faster convergence	Transition to the anchor-free mechanism
YOLOv9 [97], [113]	GELAN backbone emphasizing gradient path planning	SEAM and multiSEAM modules for fusion	Decoupled head	Anchor-free	PGI, GELAN backbone	Better generalization; improved robustness on occluded objects	New optimization techniques
YOLOv10 [99]	Enhanced CSPNet with a hybrid attention mechanism	PSA-enhanced feature extraction	Decoupled head	Anchor-free	EfficientRepV3 backbone, dual label assignment, NMS-free	Balanced performance across different domains	Focused on lightweight design
YOLOv11 [100], [114]	C3k2 module backbone for higher speed	C2PSA-enhanced feature fusion	Decoupled head	Anchor-free	Lightweight convolution modules, C2PSA attention	Enhanced lightweight structure with competitive accuracy	Efficient attention to small object
YOLOv12 [103]	Area Attention and lightweight MLP backbone	R-ELAN for optimized feature fusion	-	Anchor-free	Global feature modeling	Further improvement in energy efficiency and small object detection	Integration of visual and LiDAR modalities

efficiency, remove position encoding, adjust the MLP (multi-layer perceptron) ratio, and reduce the stacking block depth.

The YOLO algorithm's overall architecture has generally remained unchanged since its development. Each method has been refined to enhance object detection performance by addressing its predecessors' shortcomings and integrating new technological advancements. Table 2 shows architectural evolution, key improvements, benefits, and evolutionary notes of YOLO versions from YOLOv1 to YOLOv12.

In practice, deploying the YOLO model to embedded devices faces many limitations. Limited computing resources, memory bandwidth, and power budgets restrict the choice of models for real-time inference. For example, platforms such as NVIDIA Jetson Nano and Xavier NX require

models with low FLOPs and minimal parameter size to avoid overheating and power consumption limitations [115]. These practical limitations must be considered together with traditional evaluation metrics such as mAP and FPS to ensure that the model can be deployed reliably and efficiently.

To provide a clear and practical comparison of major YOLO models and their lightweight variants, Table 3 presents a performance analysis focusing on key trade-offs between detection accuracy and inference speed. In addition, to reflect real-world deployment scenarios, the table incorporates benchmarked inference performance on edge devices such as NVIDIA Jetson Nano. This comparative evaluation explicitly addresses the evolving balance between accuracy and computational efficiency, which is critical for mobile and edge

TABLE 3. Performance Comparison and Trade-Off Analysis of Lightweight YOLO Variants for Mobile/Edge Computing Applications.

Model	Params (M)	FLOPs (B)	FPS	mAP@0.5(COCO)	Trade-off type
YOLOv1 [16]	35.07	32.41	45	63.40	Speed-prioritized, low accuracy
YOLOv2 [75]	50.95	30.80	67	44.00	Improved accuracy, moderate speed
YOLOv3 [76]	61.55	155.60	45	57.90	Accuracy-prioritized, slower
YOLOv4 [109]	64.00	180.30	65	65.70	Balanced accuracy and speed
YOLOv5s [116]	7.20	16.50	27(Jetson Nano)	72.70	Balanced for edge devices
YOLOv5n [116]	1.90	4.50	42(Jetson Nano)	64.10	Speed-prioritized, low accuracy
YOLOv6n [117]	4.70	11.40	34(Jetson Nano)	37.50	Balanced lightweight model
YOLOv7 [92]	7.20	11.40	161	51.40	Accuracy-prioritized, real-time capable
YOLOv8n [112]	3.20	8.70	31(Jetson Nano)	72.80	The best balance for mobile deployment
YOLOv8s [112]	11.20	28.60	20(Jetson Nano)	74.70	Accuracy-prioritized, moderate speed

TABLE 4. Representative variants of YOLO algorithms.

Variant model	Key improvements	Application scenario	Publication year	Remark
PP-YOLO [118]	Backbone enhancement, IoU Loss, CoordConv	High-accuracy tasks	2020	Big mAP gain over YOLOv3
YOLOX [119]	Anchor-free, decoupled head, SimOTA	General detection	2021	High accuracy, better convergence
YOLOF [120]	Single-level detection without FPN	High-speed detection	2021	Simplified architecture
YOLOv5-Lite [121]	Lightweight, mobile-optimized	Edge computing	2021	Smaller, faster inference
YOLO-NAS [122]	Neural Architecture Search (NAS) based YOLO model, dynamic depth/width	Real-time detection with optimized trade-offs	2023	Automatically optimized architecture for better mAP/speed balance
YOLO-World [123]	Vision-Language pretraining	Open-world detection	2024	Zero-shot detection
YOLO-UniOW [124]	Unified optimization for window-based operations, attention-guided feature fusion	Real-time multitask detection	2024	Balances accuracy and speed with lower computation cost

computing applications, particularly in autonomous driving and embedded systems. Table 3 provides a comparative analysis of the performance and trade-offs among lightweight YOLO variants objecting to mobile and edge computing environments.

C. VARIATIONS OF THE YOLO MODEL

With the continuous evolution of the YOLO family, numerous variant models have been proposed to adapt to different application scenarios, such as lightweight deployment, anchor-free detection, open-vocabulary recognition, and real-time performance optimization.

While traditional YOLO models like YOLOv1 to YOLOv12 focus primarily on improving detection accuracy and speed through architectural innovations, various derivative models further explore new directions based on the YOLO framework. These variants often incorporate novel design concepts such as decoupled detection heads, dynamic feature aggregation, advanced data augmentation strategies, and integration with vision-language models. Such improvements extend the applicability of YOLO-based detectors to

a wider range of tasks, including mobile device deployment, large-scale open-set detection, and high-speed real-time perception. Table 4 summarizes several representative YOLO variants, highlighting their key technical innovations and typical application scenarios.

D. IMPROVEMENT STRATEGIES FOR THE YOLO MODEL

With the widespread application of the YOLO series in object detection, researchers have proposed numerous enhanced models to optimize performance and address specific application requirements. These improvements focus on various aspects, including data augmentation, feature extraction, attention mechanisms, and model architecture, significantly enhancing YOLO’s effectiveness across different objects.

Based on the overall architecture of YOLO and a comprehensive review of existing literature, this study categorizes these enhancements into three main stages: input, feature extraction, and prediction. Furthermore, these improvements are classified into eight subcategories according to their specific optimization approaches: data augmentation, data preprocessing, lightweight architecture, adding attention

mechanisms, optimizing feature fusion, introducing transformers [125], improving NMS [126], and optimizing IoU loss function [80], [91], [127], [128]. Fig. 5 illustrates the classification framework of YOLO improvement strategies.

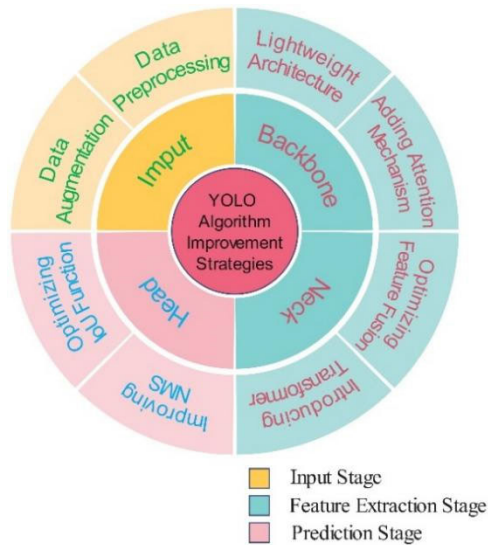


FIGURE 5. Improvement strategies for the YOLO model. These strategies are further elaborated and comparatively analyzed in Table 5, which provides corresponding methods, advantages, limitations, and applicable scenarios.

1) IMPROVEMENTS IN THE INPUT STAGE

Deep learning models rely heavily on data-driven learning, and the training effectiveness of YOLO and its improved versions is closely tied to the quality of input data. Researchers have integrated various data augmentation and preprocessing techniques into the input stage of YOLO to enhance model generalization, address data scarcity, and overcome the limitations of homogeneous datasets. These methods generate new data that align with the potential distribution of existing datasets, thereby increasing diversity and improving model performance.

Chang et al. [129] improved the YOLO model. They used image preprocessing and data augmentation techniques to highlight the key features in the training data and modified the original YOLO model by replacing the residual module with the SENet [130] module. This improvement emphasized key features and reduced the interference of low-level features, improving classification performance and thus improving detection accuracy.

Lee et al. [131] developed the YOLO-MR model, incorporating the MAML (model-agnostic meta-learning) [132] to address challenges posed by imbalanced and sparse image datasets. This approach effectively mitigated data imbalance issues, enhancing the model's robustness in diverse scenarios.

2) IMPROVEMENTS IN THE FEATURE EXTRACTION STAGE

The feature extraction stage aims to extract the basic features of the input image to support the object area prediction.

The feature extraction quality directly affects the network model's detection performance. Given its network scale, running speed, and feature extraction capabilities, scholars have proposed improvements to improve image feature extraction and feature fusion capabilities to obtain more feature information. Depending on the location of the improvement, it includes improvements to the backbone and neck. In terms of the content of the improvement, it includes lightweight structure, adding attention mechanism, optimizing feature fusion, introducing transformer, and other methods.

Fan et al. [133] developed a lightweight object detection algorithm based on YOLOv8 and LUD YOLO. The algorithm optimizes the model architecture by introducing a novel multi-scale feature fusion mechanism, thereby making it sparser and more lightweight for deployment on edge devices. Its effectiveness and superiority have been validated on the VisDrone [134], [135] dataset.

Wei et al. [136] improved the YOLOv5s network and proposed a new network structure, SED-YOLO. The network introduced a multi-scale attention mechanism in the backbone network, which reduced the computational cost while retaining key information. Moreover, the neck network utilizes an adaptive Concat module that dynamically modifies the feature fusion approach based on the image content and object features, thereby substantially enhancing the model's capacity to manage multiple objects. The mAP of the improved model on the DOTA [137] dataset is 2.4% higher than that of the original YOLOv5s.

Tian et al. [138] proposed a multi-scale Dense YOLO (MD-YOLO) for small object detection. In the feature extraction stage, the method introduced the DenseNet [139], [140] module, and AAM (adaptive attention mechanism module), which can efficiently capture image details. In addition, deep hierarchical structures such as feature extraction path and feature aggregation path are integrated into the network to effectively utilize the spatial position information in the shallow network and further improve detection accuracy. Experimental results show that the improved model achieves a mAP@0.5 of 86.2%, an F1-score of 79.1%, and an IoU of 88.1%.

Wang et al. [141] used DCN (deformable convolutional network) [142] in the backbone network based on YOLOv5, replacing the traditional convolution so that feature extraction can be more based on the shape of the detected object. The CBAM (convolutional block attention module) [143] was incorporated into the neck structure to mitigate the extraction of features from complex backgrounds by applying weights to various features while simultaneously strengthening the representation of object features. The findings indicate that the detection accuracy of the enhanced model is 2.3% higher than that of the original network.

Cheng [144] introduced the transformer structure into the YOLO v8s architecture to extract global features and used the RepConv [145] module to lower computational demands, thus balancing inference speed and detection

TABLE 5. Comparison of 8 improvement strategies.

Improvement strategies	Improvements	Advantages	Limitations	Applicable scenarios
Data augmentation [129], [149]	Rotation, flipping, scaling, cropping.	It enhances data diversity, improves model generalization, and increases robustness.	It increases training time and may introduce noise.	Insufficient data and diverse object categories.
Data preprocessing [129], [131]	Data cleaning, normalization, and standardization.	Improves data quality and enhances training efficiency.	It may negatively impact model performance.	Scenarios for standardized data processing.
Lightweight architecture [133], [150]	Utilizing lightweight backbone networks or optimizing model architecture.	Reduces computation and enhances model performance on edge devices.	It may compromise detection accuracy.	Efficient object detection in limited-resource settings.
Adding attention mechanism [136], [141]	Incorporating attention mechanisms such as the SE module and CBAM.	Enhances detection accuracy by focusing on key regions.	Increases model complexity and computational cost.	Small object detection and dense object scenarios.
Optimizing feature fusion [136], [138]	Utilizing feature pyramid networks (FPN), BiFPN.	Enhances multi-scale feature fusion.	Increases model complexity.	Multi-scale object detection.
Introducing transformer [144], [151]	Integrating transformer architecture.	Enhances the model's ability to capture global features.	High computational complexity and extended training time.	Complex scenes or small object detection.
Improving NMS [146], [148], [152], [153]	Utilizing improved methods such as Soft-NMS and adaptive NMS.	Reduces false positives.	Higher computational complexity.	It may lead to false positives and missed detections.
Optimizing IoU loss function [146], [148]	Incorporating variants such as CIoU, DIoU, and GIoU.	Enhances bounding box regression accuracy by more effectively measuring the difference between predicted and ground-truth boxes.	Requires hyperparameter tuning for optimal performance.	When high precision in bounding box regression is not required.

accuracy. It enhanced detection accuracy without sacrificing speed. However, it is worth noting that although the Transformer architecture is effective in capturing global dependencies, it usually incurs huge computational overhead, especially when applied to high-resolution inputs or resource-constrained environments. Therefore, careful architecture design and module selection are critical to ensure the practical deployment of the Transformer-enhanced YOLO model.

3) IMPROVEMENTS IN THE PREDICTION STAGE

The prediction stage transforms extracted feature information into concrete detection results, including object location, classification, and confidence scores. Research efforts in this area primarily object two aspects: 1) improving the NMS mechanism to filter and eliminate redundant bounding boxes more accurately and 2) optimizing the IoU loss function to enhance prediction accuracy by refining the constraint between ground truth and predicted bounding boxes.

Kaushal [146] proposed a novel deep learning architecture, Rapid-YOLO, based on YOLOv3. This model integrates the chaotic whale optimization (CW-NMS) [147] algorithm to replace traditional NMS, effectively mitigating false bounding box detections. Additionally, Rapid-YOLO introduces a tangent loss function to optimize confidence and classification loss, improving training efficiency and reducing errors.

Experimental results demonstrate that these enhancements significantly boost detection accuracy and precision.

Li et al. [148] developed YOLO-ACN, a detector designed to improve the detection of small objects and occluded objects. The model employs the CIoU loss function for more precise bounding box regression. Moreover, CIoU is incorporated into Soft-NMS, combined with a Gaussian model to suppress surrounding bounding boxes, resulting in more accurate selections. Experimental findings indicate that YOLO-ACN achieves high detection accuracy and efficiency, particularly for small and occluded objects.

4) COMPARISON OF IMPROVEMENT STRATEGIES

The improvement strategies span the input, feature extraction, and prediction stages, each contributing uniquely to enhancing detection accuracy and speed. Most research efforts focus on optimizing the feature extraction and prediction stages. Rather than relying on a single approach, researchers typically integrate multiple strategies to achieve superior performance. Table 5 provides a comparative summarizes the eight improvement strategies across these three stages.

In summary, since its emergence, the YOLO models have been constantly seeking a balance between detection speed and accuracy and have continued to innovate and optimize the architecture. From the early anchor frame mechanism, multi-scale detection, and feature pyramid network to the lightweight structure, attention mechanism, and Transformer

TABLE 6. Application of YOLO algorithms in vehicle object detection.

Ref.	Model-based	Key improvements	Dataset	FPS	mAP/%	mAP@0.5/%	R/%	P/%
[161]	YOLOv3-tiny	Network sparseness and quantization	nuScenes	28.00	69.79	-	82.00	91.00
[162]	YOLOv3-tiny	SPP Block, GIoU Loss, and K-means++ algorithm	COCO	49.00	52.95	-	-	-
[166]	YOLOv4	CSPBlock residual network, Leaky ReLU activation function	KITTI	68.20	82.12	72.60	-	-
[169]	YOLOv8n	Feature sharing detection head, C2f_DWR	UA-DETRAC [174]	314.10	-	97.80	94.40	94.50
[172]	YOLOv8	C2f_DCNnetwork, occlusion perceptual attention mechanism	Cityscapes	41.46	95.23	-	97.19	96.02
[173]	YOLOv7	Res3Unit, ACmix, and RFLA gauss module	UA-DETRAC	96.00	95.10	-	-	-

module in recent years, YOLO has continuously broken through its performance bottleneck while maintaining the advantages of single-stage detection. In addition, YOLO has made effective improvements in the three stages of input, feature extraction, and prediction, which not only reflects the high scalability of the YOLO architecture but also provides theoretical and technical support for subsequent actual deployment and scene adaptation.

IV. APPLICATION OF YOLO IN AUTONOMOUS DRIVING OBJECT DETECTION

Due to its efficiency and accuracy, the YOLO series of algorithms have been widely applied across various fields, including medicine, industry, real-time object detection, robotic control, autonomous driving, agriculture, video surveillance, and security [154], [155], [156], [157], [158], [159]. In autonomous driving, YOLO is utilized for real-time object detection and road obstacle identification, enabling the accurate and timely analysis of traffic conditions, including vehicles, pedestrians, and traffic signals. This section primarily explores the application of YOLO in object detection within autonomous driving scenarios, focusing on vehicle detection, pedestrian detection, traffic sign detection, traffic light detection, lane line detection, and multi-sensor fusion detection.

A. VEHICLE DETECTION

Vehicle detection is an essential task in autonomous driving. It provides the autonomous driving system with the location and motion information of other vehicles in the surrounding environment [160]. The YOLO algorithm can accurately and quickly detect vehicles in images. By monitoring the location and speed of surrounding vehicles in real time, the autonomous driving system can adjust its driving strategy to avoid dangerous collisions.

Chen and Lin [161] proposed an embedded real-time vehicle detection system based on an improved YOLO network. By optimizing the YOLOv3-tiny network structure and performing sparse and quantized processing on the network, the

32-bit floating-point weights were quantized to 8-bit integers, significantly improving the computational efficiency. Experiments show that the improved YOLOv3-live achieves a detection speed of 28 FPS on embedded devices, and the mAP reaches 69.79%

Jamiya and Rani [162] modified the feature extraction network in the YOLOv3-tiny model by incorporating MSE (mean squared error) [163] for bounding box regression and generalized intersection over union (GIoU). K-means++ [164], [165] clustering was employed to optimize anchor box sizes, improving detection speed and accuracy.

Liang et al. [166] designed an edge-YOLO detection system based on edge-cloud collaboration [167], [168] and a reconstructed CNN that integrates pruned feature extraction networks and compressed feature fusion networks, achieving a lightweight multi-scale object detection system with improved efficiency.

Liu et al. [169] proposed an improved YOLOv8n-based object detection method, introducing a feature-sharing detection head to enhance classification and localization capabilities. In the backbone network, they replaced two layers of the C2f module with a DWR (dilation-wise residual) [170] module to reduce computational complexity. Additionally, they optimized the detection process using a DySample upsampling [171] module, improving detection accuracy and speed.

Lei et al. [172] proposed an improved YOLOv8 algorithm that integrates an adaptive neural network into the residual matrix to enhance feature representation. Additionally, they optimized traditional convolutional structures to reduce network parameters, significantly improving detection accuracy.

Zhang et al. [173] developed an object detection model based on YOLOv7 by reconstructing the backbone network to achieve nonlinear feature fusion and incorporating a hybrid attention mechanism to enhance vehicle object focus. Their approach successfully improves detection accuracy while maintaining real-time processing capabilities.

In studying vehicle detection using the YOLO algorithm, [162] introduces the GIoU loss function to optimize bounding box regression. In contrast, [166] incorporates the leaky

ReLU activation function to improve detection accuracy. References [172] and [173] integrate attention mechanisms to enhance small object detection. Additionally, it can be observed that most model improvements focus on the feature extraction stage, with varying impacts on accuracy and speed, as summarized in Table 6.

B. PEDESTRIAN DETECTION

Pedestrian detection is equally important for autonomous driving systems, enabling vehicles to identify and avoid pedestrians. The YOLO algorithm can rapidly and accurately detect pedestrians in images, providing the autonomous driving system with information about pedestrian positions and movement trajectories. It allows the system to slow down or stop in time when encountering pedestrians, ensuring their safety. However, pedestrian detection often faces challenges like small object sizes and occlusions.

Xu et al. [175] proposed a new optimization algorithm to enhance the performance of the YOLOv5 model. By introducing a new optimization method, they replaced the C3 module with the Grey-C3 module and added a detection head. Compared to the YOLOv5s model, their approach improved pedestrian detection accuracy by 2.6%.

Xu et al. [176] restructured YOLOv3 by building the backbone network using a G-module combined with depth-wise convolution [177]. They incorporated an attention mechanism and designed a lightweight model, reducing model parameters, speeding up inference, and improving detection accuracy by 7%.

Ji et al. [178] proposed a multi-scale detection approach and a soft-CIoU loss function specifically for detecting small objects like pedestrians. They introduced an expanded field-of-perception block, demonstrating better performance than other detection models in detecting small objects.

Tang et al. [179] developed a pedestrian feature-enhanced lightweight model, PFEL Net [179], based on YOLOv8. They improved the feature extraction network and detection head, making it more suitable for edge computing and precise detection of multi-scale pedestrian objects in complex scenarios.

Lan et al. [180] modified the YOLOv2 network architecture by reducing the number of penetration layer connections from 16 to 12. This adjustment improved detection speed, while experimental results showed a reduced missed detection rate. Fig. 6 compares the missed detection rate of several pedestrian detection algorithms.

Liu et al. [181] proposed a lightweight, dense pedestrian detection model, MSCD-YOLO. Based on the YOLOv8 architecture, this model introduces a multi-scale feature fusion module (SCNeck) and a dynamic head module (DyHead), significantly enhancing pedestrian detection performance in complex scenarios.

Xue et al. [182] designed a YOLO-based multi-modal attention fusion detection algorithm. They utilized a multi-modal feature extraction module based on a com-



FIGURE 6. Comparison of the miss rate of several pedestrian detection algorithms.

pressed darknet-53 framework and a modality-weighted fusion module to integrate multi-modal features. Additionally, a dual attention module (Fig. 7) was incorporated to extract richer semantic features from low-resolution small objects, improving pedestrian detection performance.

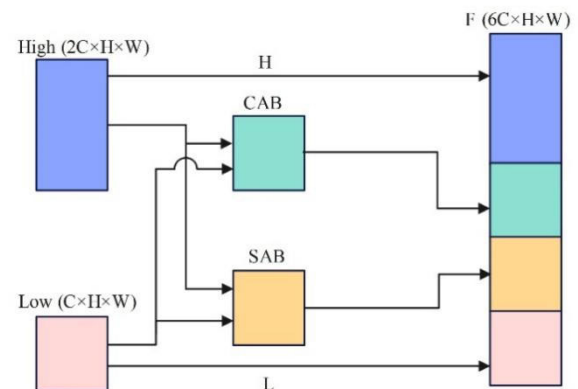


FIGURE 7. Dual attention module.

In the study of pedestrian detection using the YOLO algorithm, [175] primarily focuses on detecting small-sized objects, while [176], [178], and [179] aim to improve detection capabilities by reducing missed and false detection. Reference [182] employs multi-modal recognition to enhance pedestrian detection, as summarized in Table 7. Currently, pedestrian detection performance is relatively good. However, challenges remain in detecting occluded or small-sized pedestrians in complex and dense environments.

C. TRAFFIC SIGN DETECTION

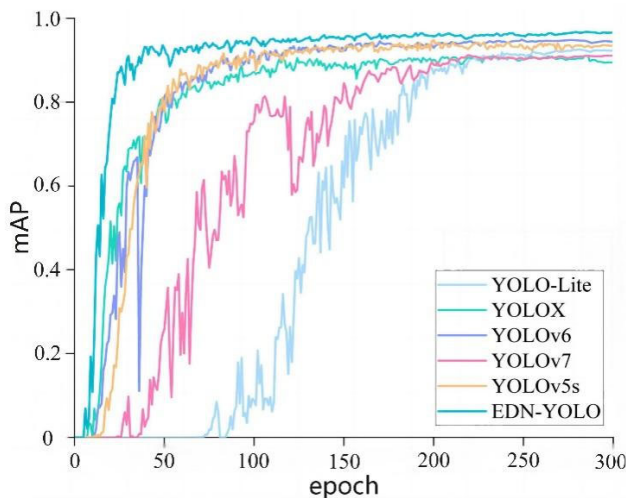
In autonomous driving scenarios, the precision of traffic sign recognition is crucial for advanced driver assistance systems and autonomous vehicles [19].

Zhao et al. [186] combined the Mamba framework with YOLO technology and proposed a new model called TSD-YOLO, which enhances the accuracy and robustness of traffic sign detection under adverse weather conditions and complex environments.

TABLE 7. Application of YOLO algorithms in pedestrian detection.

Ref.	Model-based	Key improvements	Dataset	FPS	mAP/%	mAP@0.5/%	R/%	P/%	Feature
[175]	YOLOv5	Designing Grey-C3 module.	KITTI	-	-	91.80	83.20	94.00	Small size
[176]	YOLOv3	G-Module, G-Bottleneck, depth-wise convolution.	KITTI	25.50	93.10	-	92.60	90.10	Missed detection
[178]	YOLOv4	Multi-scale contextual information, soft-CIoU loss function.	UCAS-AOD [183]	35.30	84.60	-	87.60	-	Missed detection
[179]	YOLOv8	Parallel dilated residual, channel partition selection module, lightweight shared detection head.	Cityscapes	-	-	65.60	54.40	78.00	Missed detection
[181]	YOLOv8	Introducing mobileViT, SCNeck, and dynamic head module.	Crowd-Human [184]	-	-	80.40	70.60	-	Complex scenes
[182]	YOLOv3	Multi-modal attention module, C-Darknet53 structure.	KAIST [185]	40.00	-	-	-	-	Multi-modal recognition

Sun et al. [187] optimized the YOLOv8 architecture by incorporating enhancement, reconstruction, and fusion techniques to propose a multiscale detection model (ERF-YOLO). This model demonstrated excellent detection performance, achieving a mAP of 82.4%.

**FIGURE 8.** mAP performance comparison of EDN-YOLO with other algorithms.

Han et al. [188] improved YOLOv5s and proposed the EDN-YOLO traffic sign detection method. They used the EfficientViT backbone network to improve global feature perception and decoupled the detection head to optimize classification and regression tasks. This method improved mAP by 9.6% on the TT100K dataset and effectively solved the problems of false detection and missed detection of multi-scale target detection in complex scenes. Fig. 8 shows the mAP comparison between this method and other algorithms.

Meng et al. [189] optimized the network by improving the original SPPCSPC module of YOLOv7 and designing the shuffle attention-CARAFE upsampling operator, thereby improving the detection performance of minor traffic signs and reducing the number of algorithm parameters.

Niu et al. [190] improved the backbone network based on YOLOv8 to improve the quality of feature extraction, improved the neck network to achieve multi-scale fusion, and optimized the loss function to solve the problems of traffic sign detection, small object size, and complex and scattered scenes. The model achieves the fusion of multi-scale object features while maintaining the calculation speed, significantly improving the small object detection capability.

Table 8 shows the relevant research on traffic sign detection, and it can be found that the improvement strategies are mainly concentrated in three aspects: 1) Expanding or strengthening the datasets. Reference [189] expanded the dataset of Chinese traffic signs, but the data types involved in the experiment were incomplete. The labeled and recognized data were centered around three categories: instruction signs, prohibition signs, and warning signs. 2) Improving the ability to detect small objects. Reference [188] added a small object detection layer, [189] designed the shuffle attention-CARAFE upsampling operator to improve small object detection, and [190] improved small object detection through multiscale fusion. 3) Optimizing the loss function. Reference [190] optimized the IoU loss function while adopting other optimization strategies.

D. TRAFFIC LIGHT DETECTION

Traffic lights provide essential information for road users, and their detection plays a vital role in intelligent vehicles and advanced driver assistance systems.

Garg et al. [192] fine-tuned the YOLOv8 model, created a new synthetic dataset, and trained the model. Their approach

TABLE 8. Application of YOLO algorithms in traffic sign detection.

Ref.	Model-based	Key improvements	Dataset	mAP/%	R/%	P/%	F1
[186]	YOLOv8	Visual backbone network based on state space models (SSMs), dual-branch structure.	TT-100K [191]	93.27	93.47	84.47	88.47
[187]	YOLOv8	SFE module, WFR module, and ASF module.	TT-100K	84.20	75.90	85.40	80.30
[188]	YOLOv5s	EfficientVit, decoupled detection head, and NWD loss function.	TT-100K	79.10	-	-	77.00
[189]	YOLOv7	Spatial pyramid pooling fast and cross-stage partial connection, shuffle attention CARAFE, normalized wasserstein distance.	CCTSDb	-	82.82	90.81	-
[190]	YOLOv8	Introduced C2fDWR DRB module and wise IoU.	TT-100K	-	80.00	85.20	-

demonstrated strong traffic light detection capabilities under adverse weather conditions, including rain, snow, and fog.

Bakirci [193] explored the application of YOLOv8 in intelligent transportation systems (ITS), specifically improving traffic light detection. By optimizing feature extraction and detection processes, YOLOv8 performed exceptionally well in traffic monitoring tasks, effectively recognizing traffic light states and providing real-time support for autonomous driving systems.

Choi et al. [194] in order to solve the problem of traffic light recognition for uncrewed delivery vehicles, the performance and efficiency of traffic light state recognition were significantly improved by improving the network architecture of the TLSR module and introducing a ratio-preserving zero-filling method. At the same time, by optimizing the parameters of the TLD module, efficient traffic light detection was achieved in real-time scenarios.

Wei et al. [195] developed the SCCA-YOLO model based on YOLOv8. The model used spatial and channel collaborative attention mechanisms and was tested and compared on self-made and public datasets. It significantly improved the detection accuracy and efficiency of small autonomous driving perception system objects in remote highway scenarios.

Ngoc et al. [196] proposed an end-to-end traffic light detection model based on YOLO8 and evaluated it in a simulation environment. The results show that the model outperforms other YOLO models in traffic light detection.

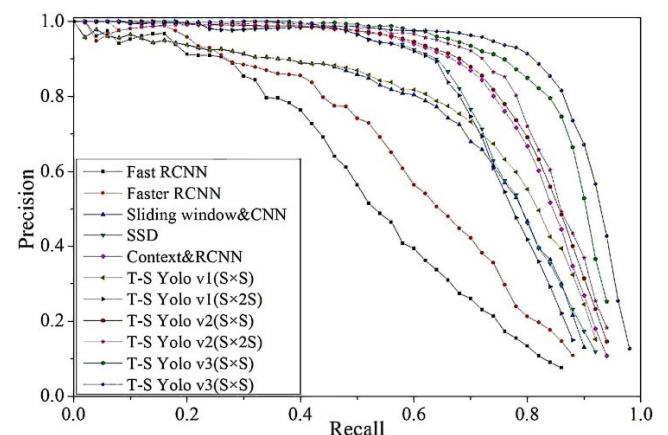
Table 9 shows the related research on traffic light detection. Reference [192] created a synthetic dataset to improve the accuracy of input data, [193] optimized the feature extraction and detection process, [194] improved the network architecture, [195] added an attention mechanism, and [196] evaluated in a simulated environment. The table shows that most of these studies used self-made or synthetic datasets. In traffic light detection, the public dataset cannot provide enough labels to train deep neural networks, which requires further research.

E. LANE LINE DETECTION

Efficient lane line detection is a vital part of autonomous vehicles' road environment perception module, and its accuracy directly impacts their safety.

Liu et al. [197] incorporated the Faster-Net lightweight network into YOLOv3 to reduce computational complexity. They also integrated ECA (efficient channel attention) [198] into the backbone network to improve accuracy and introduced SIOU to enhance robustness.

Zhang et al. [199] modified the structural parameters of the YOLOv3 algorithm and adopted an adaptive edge detection algorithm based on the Canny [200] operator. This method repositions lanes detected by the first-stage model. It uses the processed images as labeled data for the second-stage model training, significantly improving lane detection accuracy in complex environments. Fig. 9 presents this model's precision-recall (P-R) curves and other algorithms on the KITTI dataset. Detection algorithms based on the YOLO series exhibit notable advantages in lane detection, particularly the $S \times 2S$ structured model based on YOLOv3, which has demonstrated outstanding performance.

**FIGURE 9.** P-R curves of algorithms on the KITTI dataset.

Li et al. [202] proposed a lane detection method based on the reformer [203] model. In order to solve the problems of high computational complexity and complex optimization of the traditional transformer model, they introduced the local sensitive hashing attention mechanism and reversible structure to reduce the complexity. At the same time, they adopted the mish activation function and Huber loss function [204],

TABLE 9. Application of YOLO algorithms in traffic light detection.

Ref.	Model-based	Key improvements	Dataset	mAP@0.5/%	R/%	P/%	F1	FPS
[192]	YOLOv8	Synthetic datasets, innovative fine-tuning, and training procedures.	Synthetic datasets	50.40	-	86.00	57.60	-
[193]	YOLOv8	Multi-scale feature fusion and dataset optimization.	Self-made datasets	-	61.90	80.30	67.50	83.30
[194]	YOLO	Traffic light status recognition (TLRS) module and traffic light detection (TLD) module.	Self-made datasets	88.30	-	-	-	93.30
[195]	YOLOv8	Spatial channel collaborative attention mechanism integrated Ghos module.	Self-made datasets	84.40	73.30	88.70	-	-
[196]	YOLOv8	Data augmentation.	CinTA_v2, GathoTF	91.90	84.30	93.20	89.50	-

[205] to improve the training efficiency and detection accuracy. Experiments show that this method performs well in actual scenarios and has high practicality and promotion value.

Huu et al. [206] presented a YOLO-based algorithm for lane and obstacle detection. Lane detection was achieved through image preprocessing and sliding window [207] mechanism, with an accuracy of 97.91%. The algorithm performs well in accuracy and real-time performance, has strong noise resistance, and is suitable for complex road conditions and advanced network environments.

Deng et al. [208] proposed a joint vehicle and lane detection algorithm based on MobileNetV3 [209]. Through the encoder-decoder structure and multitask learning, the feature extraction capability is improved by combining ASPP (atrous spatial pyramid pooling) and FPN, and the CBAM attention mechanism is introduced to optimize YOLOv4. At the same time, the asymmetric SegNet [210] branch is used to achieve lane segmentation. This approach's vehicle detection speed is 1.7 times faster than YOLOv4's, and its lane segmentation speed is 3.2 times quicker than SegNet's.

While studying lane lines, [197] proposed a lightweight network and attention mechanism to improve the real-time accuracy of detection. Reference [199] proposed an automatic labeling algorithm for labeled images, which enriched the data set and improved the labeling efficiency. Reference [202] introduced the reformer model to improve the accuracy of lane detection. Reference [206] used image preprocessing and sliding window methods to improve lane line detection's accuracy and real-time performance. Reference [208] used the lightweight network MobileNetV3 to improve the network detection speed or modified the network density to make the algorithm more suitable for lane line detection. However, there are still limitations in lane line missed detection in large slopes and complex scenes, which needs further research. Table 10 shows the performance of the above lane line detection cases.

F. MULTI-SENSOR FUSION DETECTION

Multi-sensor fusion technology is of great significance in autonomous driving. It improves the accuracy and robust-

ness of object detection by effectively integrating data from different sensors (such as cameras, lidar, and millimeter-wave radar). Cameras are prone to false detection or missed detection under complex lighting conditions such as intense light and backlight, while lidar has strong three-dimensional shape and distance perception capabilities. After the fusion of the two, the YOLO algorithm can identify objects more accurately, especially when detecting small objects at long distances. In addition, millimeter-wave radar has a high detection accuracy for moving objects. Combined with cameras, it can enhance YOLO's perception of dynamic traffic scenes and improve its detection capabilities for fast-moving vehicles and pedestrians.

Multi-sensor fusion usually includes data-level fusion, feature-level fusion, and decision-level fusion [213]. These fusion methods can improve the performance of autonomous driving systems in complex environments.

Islam et al. [214] used YOLOv5 as the base detector and significantly improved the pedestrian detection performance by fusing camera and LiDAR data. This method reduced the pedestrian false detection rate from 41.945% to 33.435% while maintaining real-time performance.

Shi et al. [215] proposed the CL-FusionBEV method for camera and lidar data fusion in autonomous driving. They performed 3D object detection from a bird's-eye view, and achieved a mAP of 73.3% on the nuScenes dataset, with vehicle and pedestrian detection accuracies of 89% and 90.7%, respectively.

Jin et al. [216] proposed a hybrid object detection model based on feature-level camera-millimeter wave radar fusion for object detection in autonomous driving scenarios. The model was tested on the nuScenes dataset, and the average precision (mAP) of the model was improved by 1.4%.

Rakotoniaina [217] used the occupancy grid algorithm to fuse lidar and radar data with the YOLO detection results. On the nuScenes dataset, the method achieved a detection accuracy of 68.7% mAP, effectively solving the problems of object occlusion and data association.

Carranza-García et al. [218] proposed an autonomous driving object detection method based on depth completion and camera-LiDAR fusion. By designing an efficient LiDAR depth completion network, sparse LiDAR data is converted

TABLE 10. Application of YOLO algorithms in traffic line detection.

Ref.	Model-based	Key improvements	Dataset	mAP@0.5/%	R/%	P/%	FPS
[197]	YOLOv3	C3_Faster module, ECA mechanism, SIoU.	CULane [211]	95.10	-	-	-
[199]	YOLOv3	The two-level network automatically learns lane features and divides the image into S×2S grids.	KITTI, Caltech-lanes [201]	88.39	-	-	-
[202]	-	Sigmoid, ReLU, and Mish.	TuSimple [212]	-	-	97.65	-
[206]	YOLO	ROI extraction, inverse perspective transformation, and sliding window.	TuSimple	-	-	89.90	-
[208]	YOLOv4	Convolutional block attention module.	BDD100K, KITTI	63.70	62.20	79.20	28.70

into dense depth maps and fused with camera images, significantly improving the performance of object detection.

Murendeni et al. [219] improved the YOLOv4 architecture and achieved an efficient fusion of multi-modal sensors by fusing RGB cameras and lidar point cloud data. The results show that the model achieves a mAP of 0.85 while maintaining real-time performance, which is significantly better than YOLOv4 detection.

Nie et al. [220] studied how to use stereo cameras and LiDAR data for 3D object detection. By fusing multi-modal sensor data, an efficient 3D object detection method was proposed, which significantly improved the detection accuracy and robustness. Fig. 10 shows the previous fusion method and the innovative fusion framework.

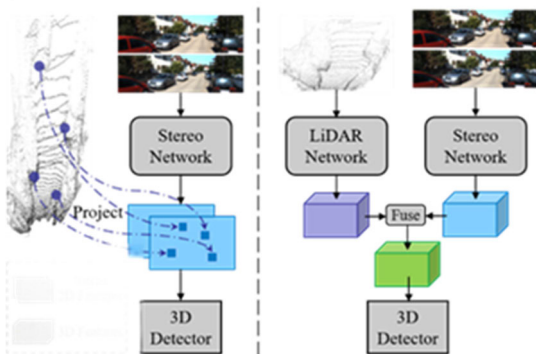


FIGURE 10. Previous and innovative fusion frameworks [220].

Table 11 summarizes some of the multi-sensor fusion approaches for autonomous driving, highlighting the diversity of sensor combinations, fusion strategies, and the resulting detection performance.

In addition to academic advancements, YOLO-based object detection algorithms have seen practical adoption in the autonomous driving industry. Companies such as Tesla utilize real-time vision systems inspired by single-stage detectors like YOLO to enhance perception modules in their FSD (full self-driving) platforms [221]. Waymo integrates deep learning-based vision detection frameworks, often adopting optimized versions of one-stage detectors within their multi-sensor fusion stacks [222], [223]. Fur-

thermore, open-source autonomous driving initiatives like Apollo [224], [225] and Autoware [226], [227] have incorporated YOLO variants into their perception pipelines, highlighting YOLO’s real-world applicability due to its balance of inference speed and accuracy. These deployments underscore the growing relevance of YOLO models in bridging the gap between academic research and industrial-scale autonomous driving systems.

In summary, the YOLO algorithm has been widely used in multi-class object detection tasks in autonomous driving due to its excellent real-time performance and accuracy, including key objects such as vehicles, pedestrians, traffic signs, traffic lights, and lane lines. In specific applications, different studies have effectively improved YOLO’s perception of small objects, occluded objects, and complex scenes through various improvement strategies. At the same time, the multi-sensor fusion strategy further enhances YOLO’s detection robustness and accuracy performance. Various fusion methods have achieved good performance in different data sets, verifying the actual deployment potential of the YOLO algorithm. Overall, YOLO has not only continued to make breakthroughs in academic research but also demonstrated substantial engineering value and implementation capabilities in the industry.

V. CHALLENGES AND FUTURE DIRECTIONS

Although the YOLO model series has made significant progress in autonomous driving object detection, they still face many challenges in complex and demanding scenarios such as autonomous driving. From the application analysis in section IV, it can be seen that small-sized object detection has missed detections and false detections, the multi-modal object detection capability is not high, the detection speed and accuracy are difficult to balance in complex environments, the model structure is not concise enough, and the data set is not perfect. In response to these challenges, this article proposes the following development directions.

- Enhancing Small Object Detection

Small objects in autonomous driving scenarios, such as pedestrians and traffic signs in the distance, are crucial but are often missed or misdetected due to resolution and feature weakening. In the future, we can effectively improve the abil-

TABLE 11. Performance comparison of different fusion strategies in autonomous driving.

Ref.	Model structure	Fusion type	Sensor	Dataset	mAP (%)	Key contributions
[214]	Fused-YOLOv5	decision-level	Camera + LiDAR	Waymo	-	Significantly improve pedestrian detection performance
[215]	CL-FusionBEV	feature-level	Camera + LiDAR	nuScenes	73.30	Strong robustness
[216]	YOLO-based Hybrid	feature-level	Camera + millimeter wave radar	nuScenes	47.80	Improved accuracy
[217]	YOLO + OGMap	decision-level	LiDAR + radar	nuScenes	68.70	Solving occlusion and data association problems
[218]	YOLO + LDCNet	feature-level	Camera + LiDAR	Waymo	60.50	Adapt to various detection architectures
[219]	YOLOv4 + 3D Head	feature-level	Camera + LiDAR	KITTI/nuScenes	85.00	Multi-task loss
[220]	YOLO + BEV	feature-level	Stereo camera + LiDAR	KITTI	86.90	Stereo vision and point cloud fusion framework

ity to detect small objects by optimizing the feature extraction network, designing more sophisticated anchor frames, and combining context perception mechanisms.

- *Multi-scale Object Detection*

Autonomous driving vehicles need to operate stably in various complex environments, including extreme weather, lighting changes, occlusion. The multi-scale detection performance of the algorithm is improved through multi-scale training and object enhancement, and the performance of the YOLO algorithm in intelligent driving scenarios is optimized.

- *Multi-modal Information Fusion*

Multi-modal information fusion is the fusion of sound, image, radar and other data as input data for object detection. By fusing multi-modal data, it is expected to alleviate the performance degradation problem of a single sensor in harsh environments. In the future, research on the combination of YOLO and multi-modal fusion framework should be strengthened to enhance the comprehensive perception capability of the system.

- *Lightweight and Efficient Reasoning*

In actual deployment scenarios, the lightweight design and inference speed of the detection algorithm are crucial. The algorithm complexity and computational cost are reduced by reducing the number of convolutional layers, reducing the number of channels, and using lightweight activation functions. The feature extraction capabilities of large models are transferred to small models by combining knowledge distillation technology. At the same time, model pruning and quantization technology are used to reduce memory usage and computing requirements, improve inference speed, and enable the YOLO algorithm to respond quickly to traffic changes.

- *Diverse Datasets and Model Generalization*

Existing datasets still have deficiencies in scene diversity, environmental complexity, and label granularity. Use virtual simulation technology to generate rare or special event data, develop diversified datasets, and achieve cross-scenario and cross-task capability improvement through transfer learning,

thereby enhancing the generalization ability of the model in complex traffic scenarios.

VI. CONCLUSION

This paper deeply explores the YOLO series of algorithms and their applications in the field of autonomous driving object detection. Combining the evolution process from YOLOv1 to YOLOv12, it analyzes its continuous optimization trends in network structure, detection accuracy, and inference speed. At the same time, it summarizes the current mainstream eight types of improvement strategies, including data augmentation, data preprocessing, lightweight structure, adding attention mechanisms, optimizing feature fusion, introducing transformers, improving NMS, and optimizing IoU loss function, showing the diversified development trend of the YOLO algorithm in performance improvement.

In practical applications, the YOLO series demonstrates good real-time performance and accuracy in key scenarios of autonomous driving, such as vehicle detection, pedestrian recognition, and traffic sign detection. However, it also exposes prominent problems such as insufficient detection of small objects and difficulty in balancing speed and accuracy. To this end, this paper proposes future research directions, including further improving small object detection capabilities, introducing multi-scale object detection, exploring multi-modal information fusion, further lightweighting network structures, and developing diverse data sets, which provides a useful reference for further development of the YOLO algorithm in autonomous driving.

Although this paper systematically reviews the evolution of YOLO algorithms, their improvement strategies, and applications in autonomous driving, certain limitations remain. First, the experimental settings of different enhancement methods are not unified, making it difficult to directly compare their effectiveness. Second, while multi-sensor fusion is discussed, the paper lacks an in-depth analysis of fusion mechanisms and associated challenges. Lastly, there is no systematic evaluation of the long-term performance, generalization, and energy efficiency of YOLO models in real-world deployment.

Future research should focus on unified benchmarking frameworks, optimization of multi-modal fusion, and deployment studies on edge devices.

REFERENCES

- [1] J. Zhao, W. Zhao, B. Deng, Z. Wang, F. Zhang, W. Zheng, W. Cao, J. Nan, Y. Lian, and A. F. Burke, "Autonomous driving system: A comprehensive survey," *Expert Syst. Appl.*, vol. 242, May 2024, Art. no. 122836, doi: [10.1016/j.eswa.2023.122836](#).
- [2] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "End-to-end autonomous driving: Challenges and frontiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 10164–10183, Dec. 2024, doi: [10.1109/TPAMI.2024.3435937](#).
- [3] T. Li, J. Ruan, and K. Zhang, "The investigation of reinforcement learning-based end-to-end decision-making algorithms for autonomous driving on the road with consecutive sharp turns," *Green Energy Intell. Transp.*, vol. 2025, Mar. 2025, Art. no. 100288, doi: [10.1016/j.geits.2025.100288](#).
- [4] K. D. Kusano, J. M. Scanlon, Y.-H. Chen, T. L. McMurry, R. Chen, T. Gode, and T. Victor, "Comparison of waymo rider-only crash data to human benchmarks at 7.1 million miles," *Traffic Injury Prevention*, vol. 25, no. suppl, pp. S66–S77, Nov. 2024.
- [5] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, Feb. 2020, doi: [10.1007/s11263-019-01247-4](#).
- [6] M. Shoman, D. Wang, A. Aboah, and M. Abdel-Aty, "Enhancing traffic safety with parallel dense video captioning for end-to-end event analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2024, pp. 7125–7133, doi: [10.1109/CVPRW63382.2024.00707](#).
- [7] Y. Zheng and H. Zhang, "Video analysis in sports by lightweight object detection network under the background of sports industry development," *Comput. Intell. Neurosci.*, vol. 2022, Aug. 2022, Art. no. 3844770, doi: [10.1155/2022/3844770](#).
- [8] Z. Song, L. Liu, F. Jia, Y. Luo, C. Jia, G. Zhang, L. Yang, and L. Wang, "Robustness-aware 3D object detection in autonomous driving: A review and outlook," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 11, pp. 15407–15436, Nov. 2024, doi: [10.1109/TITS.2024.3439557](#).
- [9] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. Paixão, F. Mutz, L. Veronese, T. Oliveira-Santos, and A. De Souza, "Self-driving cars: A survey," 2019, *arXiv:1901.04407*.
- [10] S. Xiang, A. Xie, M. Ye, X. Yan, X. Han, H. Niu, Q. Li, and H. Huang, "Autonomous eVTOL: A summary of researches and challenges," *Green Energy Intell. Transp.*, vol. 3, no. 1, Feb. 2024, Art. no. 100140, doi: [10.1016/j.geits.2023.100140](#).
- [11] A. Pravalika, M. F. Hashmi, and A. Gupta, "Deep learning frontiers in 3D object detection: A comprehensive review for autonomous driving," *IEEE Access*, vol. 12, pp. 173936–173980, 2024, doi: [10.1109/ACCESS.2024.3456893](#).
- [12] J. Mao, S. Shi, X. Wang, and H. Li, "3D object detection for autonomous driving: A comprehensive survey," *Int. J. Comput. Vis.*, vol. 131, no. 8, pp. 1909–1963, Aug. 2023, doi: [10.1007/s11263-023-01790-1](#).
- [13] Y. Wang and J. Ye, "An overview of 3D object detection," 2020, *arXiv:2010.15614*.
- [14] Preeti and C. Rana, "Artificial intelligence based object detection and traffic prediction by autonomous vehicles—A review," *Expert Syst. Appl.*, vol. 255, Dec. 2024, Art. no. 124664, doi: [10.1016/j.eswa.2024.124664](#).
- [15] C. Gheorghe, M. Duguleanu, R. G. Boboc, and C. C. Postelnicu, "Analyzing real-time object detection with YOLO algorithm in automotive applications: A review," *Comput. Model. Eng. Sci.*, vol. 141, no. 3, pp. 1939–1981, 2024, doi: [10.32604/cmes.2024.054735](#).
- [16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 779–788, doi: [10.1109/CVPR.2016.91](#).
- [17] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: Challenges, architectural successors, datasets and applications," *Multimedia Tools Appl.*, vol. 82, no. 6, pp. 9243–9275, Mar. 2023, doi: [10.1007/s11042-022-13644-y](#).
- [18] J. Ruan, H. Cui, Y. Huang, T. Li, C. Wu, and K. Zhang, "A review of occluded objects detection in real complex scenarios for autonomous driving," *Green Energy Intell. Transp.*, vol. 2, no. 3, Jun. 2023, Art. no. 100092, doi: [10.1016/j.geits.2023.100092](#).
- [19] M. Flores-Calero, C. A. Astudillo, D. Guevara, J. Maza, B. S. Lita, B. Defaz, J. S. Ante, D. Zabala-Blanco, and J. M. A. Moreno, "Traffic sign detection and recognition using YOLO object detection algorithm: A systematic review," *Mathematics*, vol. 12, no. 2, p. 297, Jan. 2024, doi: [10.3390/math12020297](#).
- [20] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proc. IEEE*, vol. 111, no. 3, pp. 257–276, Mar. 2023, doi: [10.1109/JPROC.2023.3238524](#).
- [21] Y.-Q. Wang, "An analysis of the viola-jones face detection algorithm," *Image Process. Line*, vol. 4, pp. 128–148, Jun. 2014, doi: [10.5201/ipol.2014.104](#).
- [22] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. CVPR*, vol. 1, Kauai, HI, USA, Jun. 2001, doi: [10.1109/CVPR.2001.990517](#).
- [23] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR05)*, vol. 1, San Diego, CA, USA, Jun. 2005, pp. 886–893, doi: [10.1109/CVPR.2005.177](#).
- [24] A. Albiol, D. Monzo, A. Martin, J. Sastre, and A. Albiol, "Face recognition using HOG-EBGM," *Pattern Recognit. Lett.*, vol. 29, no. 10, pp. 1537–1543, Jul. 2008, doi: [10.1016/j.patrec.2008.03.017](#).
- [25] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
- [26] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Anchorage, AK, USA, Jun. 2008, pp. 1–8, doi: [10.1109/CVPR.2008.4587597](#).
- [27] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 580–587, doi: [10.1109/CVPR.2014.81](#).
- [28] L. Du, R. Zhang, and X. Wang, "Overview of two-stage object detection algorithms," *J. Phys., Conf. Ser.*, vol. 1544, no. 1, May 2020, Art. no. 012033, doi: [10.1088/1742-6596/1544/1/012033](#).
- [29] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Computer Vision—ECCV*, vol. 2014, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham, Switzerland: Springer, 2014, pp. 346–361, doi: [10.1007/978-3-319-10578-9_23](#).
- [30] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
- [31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: [10.1109/TPAMI.2016.2577031](#).
- [32] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. 30th Conf. Neural Inf. Process. Syst.*, Jan. 2016, pp. 1–9.
- [33] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
- [34] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A review of YOLO algorithm developments," *Proc. Comput. Sci.*, vol. 199, pp. 1066–1073, Jun. 2022, doi: [10.1016/j.procs.2022.01.135](#).
- [35] A. Baccouche, B. Garcia-Zapirain, Y. Zheng, and A. S. Elmaghraby, "Early detection and classification of abnormality in prior mammograms using image-to-image translation and YOLO techniques," *Comput. Methods Programs Biomed.*, vol. 221, Jun. 2022, Art. no. 106884.
- [36] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*, Cham, Switzerland: Springer, Oct. 2016, pp. 21–37, doi: [10.1007/978-3-319-46448-0_2](#).
- [37] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2017, *arXiv:1701.06659*.
- [38] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

- [39] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," 2020, *arXiv:2010.04159*.
- [40] P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, L. Li, Z. Yuan, C. Wang, and P. Luo, "Sparse R-CNN: End-to-end object detection with learnable proposals," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 14449–14458, doi: [10.1109/CVPR46437.2021.01422](https://doi.org/10.1109/CVPR46437.2021.01422).
- [41] Y. Liu, J. Cao, C. Liu, K. Ding, and L. Jin, "Datasets for large language models: A comprehensive survey," *Research Square*, Feb. 2024, doi: [10.21203/rs.3.rs-3996137/v1](https://doi.org/10.21203/rs.3.rs-3996137/v1).
- [42] J. Kaur and W. Singh, "Tools, techniques, datasets and application areas for object detection in an image: A review," *Multimedia Tools Appl.*, vol. 81, no. 27, pp. 38297–38351, Nov. 2022, doi: [10.1007/s11042-022-13153-y](https://doi.org/10.1007/s11042-022-13153-y).
- [43] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: [10.1007/s11263-009-0275-4](https://doi.org/10.1007/s11263-009-0275-4).
- [44] M. Everingham, A. Zisserman, and C. K. I. Williams, "The pascal visual object classes challenge 2006 (VOC2006) results," PASCAL Netw. Excellence, Univ. Oxford, Tech. Rep., 2006. [Online]. Available: <http://homepages.inf.ed.ac.uk/ckiwi/postscript/voc2006res.pdf>
- [45] E. Zimmermann, J. Szeto, J. Pasquero, and F. Ratle, "Benchmarking a benchmark: How reliable is MS-COCO?" 2023, *arXiv:2311.02709*.
- [46] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. Lawrence Zitnick, and P. Dollár, "Microsoft COCO: Common objects in context," 2014, *arXiv:1405.0312*.
- [47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255, doi: [10.1109/CVPR.2009.5206848](https://doi.org/10.1109/CVPR.2009.5206848).
- [48] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013, doi: [10.1177/0278364913491297](https://doi.org/10.1177/0278364913491297).
- [49] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361, doi: [10.1109/CVPR.2012.6248074](https://doi.org/10.1109/CVPR.2012.6248074).
- [50] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Mallocci, A. Kolesnikov, T. Duerig, and V. Ferrari, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," *Int. J. Comput. Vis.*, vol. 128, no. 7, pp. 1956–1981, Jul. 2020, doi: [10.1007/s11263-020-01316-z](https://doi.org/10.1007/s11263-020-01316-z).
- [51] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "NuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 11618–11628, doi: [10.1109/CVPR42600.2020.01164](https://doi.org/10.1109/CVPR42600.2020.01164).
- [52] K. Tian, J. Mao, Y. Zhang, J. Jiang, Y. Zhou, and Z. Tu, "NuScenes-SpatialQA: A spatial understanding and reasoning benchmark for vision-language models in autonomous driving," 2025, *arXiv:2504.03164*.
- [53] Y. Inoue, Y. Yada, K. Tanahashi, and Y. Yamaguchi, "NuScenes-MQA: Integrated evaluation of captions and QA for autonomous driving datasets using markup annotations," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Waikoloa, HI, USA, Jan. 2024, pp. 930–938, doi: [10.1109/wacvw60836.2024.00104](https://doi.org/10.1109/wacvw60836.2024.00104).
- [54] T. Qian, J. Chen, L. Zhuo, Y. Jiao, and Y. Jiang, "NuScenes-QA: A multi-modal visual question answering benchmark for autonomous driving scenario," in *Proc. AAAI*, Jan. 2023, vol. 38, no. 5, pp. 4542–4550.
- [55] P. Sun et al., "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 2443–2451, doi: [10.1109/CVPR42600.2020.00252](https://doi.org/10.1109/CVPR42600.2020.00252).
- [56] X. Hu, Z. Zheng, D. Chen, X. Zhang, and J. Sun, "Processing, assessing, and enhancing the waymo autonomous vehicle open dataset for driving behavior research," *Transp. Res. C. Emerg. Technol.*, vol. 134, Jan. 2022, Art. no. 103490, doi: [10.1016/j.trc.2021.103490](https://doi.org/10.1016/j.trc.2021.103490).
- [57] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, "Argoverse: 3D tracking and forecasting with rich maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 8740–8749, doi: [10.1109/CVPR.2019.00895](https://doi.org/10.1109/CVPR.2019.00895).
- [58] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. Kaesemodel Pontes, D. Ramanan, P. Carr, and J. Hays, "Argoverse 2: Next generation datasets for self-driving perception and forecasting," 2023, *arXiv:2301.00493*.
- [59] P. Xiao, Z. Shao, S. Hao, Z. Zhang, X. Chai, J. Jiao, Z. Li, J. Wu, K. Sun, K. Jiang, Y. Wang, and D. Yang, "PandaSet: Advanced sensor suite dataset for autonomous driving," in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, Sep. 2021, pp. 3095–3101, doi: [10.1109/ITSC48978.2021.9565009](https://doi.org/10.1109/ITSC48978.2021.9565009).
- [60] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "BDD100K: A diverse driving dataset for heterogeneous multitask learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 2633–2642, doi: [10.1109/CVPR42600.2020.00271](https://doi.org/10.1109/CVPR42600.2020.00271).
- [61] S. Zhang, Y. Xie, J. Wan, H. Xia, S. Z. Li, and G. Guo, "WiderPerson: A diverse dataset for dense pedestrian detection in the wild," 2019, *arXiv:1909.12118*.
- [62] M. B. Jensen, M. P. Philipsen, A. Møgelmoose, T. B. Moeslund, and M. M. Trivedi, "Vision for looking at traffic lights: Issues, survey, and perspectives," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 1800–1815, Jul. 2016, doi: [10.1109/ITITS.2015.2509509](https://doi.org/10.1109/ITITS.2015.2509509).
- [63] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 3213–3223, doi: [10.1109/CVPR.2016.350](https://doi.org/10.1109/CVPR.2016.350).
- [64] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang, "The ApolloScape dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Salt Lake City, UT, USA, Jun. 2018, pp. 1067–10676, doi: [10.1109/CVPRW.2018.00141](https://doi.org/10.1109/CVPRW.2018.00141).
- [65] W. Liu, Z. Li, S. Zhang, T. Qin, and J. Zhao, "Bud-YOLOv8s: A potato bud-eye-detection algorithm based on improved YOLOv8s," *Electronics*, vol. 13, no. 13, p. 2541, Jun. 2024, doi: [10.3390/electronics13132541](https://doi.org/10.3390/electronics13132541).
- [66] H. Zhu, H. Wei, B. Li, X. Yuan, and N. Kehtarnavaz, "A review of video object detection: Datasets, metrics and methods," *Appl. Sci.*, vol. 10, no. 21, p. 7834, Nov. 2020, doi: [10.3390/app10217834](https://doi.org/10.3390/app10217834).
- [67] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A survey on performance metrics for object-detection algorithms," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Niterói, Brazil, Jul. 2020, pp. 237–242, doi: [10.1109/iwssip48289.2020.9145130](https://doi.org/10.1109/iwssip48289.2020.9145130).
- [68] N. Jegham, C. Y. Koh, M. Abdelatti, and A. Hendawi, "YOLO evolution: A comprehensive benchmark and architectural review of YOLOv12, YOLOv11, and their previous versions," 2024, *arXiv:2411.00201*.
- [69] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, "A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Mach. Learn. Knowl. Extraction*, vol. 5, no. 4, pp. 1680–1716, Nov. 2023, doi: [10.3390/make5040083](https://doi.org/10.3390/make5040083).
- [70] M. Hussain, "YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection," *Machines*, vol. 11, no. 7, p. 677, Jun. 2023, doi: [10.3390/machines11070677](https://doi.org/10.3390/machines11070677).
- [71] M. A. R. Alif and M. Hussain, "YOLOv1 to YOLOv10: A comprehensive review of YOLO variants and their application in the agricultural domain," 2024, *arXiv:2406.10139*.
- [72] C.-Y. Wang and H.-Y.-M. Liao, "YOLOv1 to YOLOv10: The fastest and most accurate real-time object detection systems," *APSIPA Trans. Signal Inf. Process.*, vol. 13, no. e29, pp. 1–38, 2024, doi: [10.1561/116.20240058](https://doi.org/10.1561/116.20240058).
- [73] Q. Ma, X. Fan, H. Chen, and T. Li, "IRE-YOLO: Infrared weak target detection algorithm based on the fusion of multi-scale receptive fields and efficient convolution," *J. Supercomput.*, vol. 81, no. 4, pp. 558–577, Feb. 2025, doi: [10.1007/s11227-025-07057-5](https://doi.org/10.1007/s11227-025-07057-5).
- [74] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, "A survey on modern trainable activation functions," *Neural Netw.*, vol. 138, pp. 14–32, Jun. 2021, doi: [10.1016/j.neunet.2021.01.026](https://doi.org/10.1016/j.neunet.2021.01.026).
- [75] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6517–6525, doi: [10.1109/CVPR.2017.690](https://doi.org/10.1109/CVPR.2017.690).
- [76] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
- [77] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 936–944, doi: [10.1109/CVPR.2017.106](https://doi.org/10.1109/CVPR.2017.106).

- [78] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
- [79] D. Zhou, J. Fang, X. Song, C. Guan, J. Yin, Y. Dai, and R. Yang, "IoU loss for 2D/3D object detection," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2019, pp. 85–94, doi: [10.1109/3DV.2019.00019](https://doi.org/10.1109/3DV.2019.00019).
- [80] Y.-F. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, and T. Tan, "Focal and efficient IOU loss for accurate bounding box regression," *Neurocomputing*, vol. 506, pp. 146–157, Sep. 2022, doi: [10.1016/j.neucom.2022.07.042](https://doi.org/10.1016/j.neucom.2022.07.042).
- [81] D. Misra, "Mish: A self regularized non-monotonic activation function," 2019, *arXiv:1908.08681*.
- [82] C.-Y. Wang, H.-Y. Mark Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Seattle, WA, USA, Jun. 2020, pp. 1571–1580, doi: [10.1109/CVPRW50498.2020.00203](https://doi.org/10.1109/CVPRW50498.2020.00203).
- [83] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768.
- [84] G. Jocher. (2020). *YOLOv5*. [Online]. Available: <https://github.com/ultralytics/yolov5>
- [85] S. K. Jaiswal and R. Agrawal, "A comprehensive review of YOLOv5: Advances in real-time object detection," *Int. J. Innov. Res. Comput. Sci. Technol.*, vol. 12, no. 3, pp. 75–80, May 2024, doi: [10.55524/ijrcst.2024.12.3.12](https://doi.org/10.55524/ijrcst.2024.12.3.12).
- [86] A. Paszke, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, BC, Canada, Jan. 2019, pp. 8024–8035.
- [87] S. Gao, M. Gao, and Z. Wei, "MCF-YOLOv5: A small target detection algorithm based on multi-scale feature fusion improved YOLOv5," *Information*, vol. 15, no. 5, p. 285, May 2024, doi: [10.3390/info15050285](https://doi.org/10.3390/info15050285).
- [88] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie, Y. Li, B. Zhang, Y. Liang, L. Zhou, X. Xu, X. Chu, X. Wei, and X. Wei, "YOLOv6: A single-stage object detection framework for industrial applications," 2022, *arXiv:2209.02976*.
- [89] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun, "RepVGG: Making VGG-style ConvNets great again," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 13728–13737, doi: [10.1109/CVPR46437.2021.01352](https://doi.org/10.1109/CVPR46437.2021.01352).
- [90] Z. Zhang and M. R. Sabuncu, "Self-distillation as instance-specific label smoothing," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Dec. 2022, pp. 2184–2195.
- [91] S. Du, B. Zhang, and P. Zhang, "Scale-sensitive IOU loss: An improved regression loss function in remote sensing object detection," *IEEE Access*, vol. 9, pp. 141258–141272, 2021, doi: [10.1109/ACCESS.2021.3119562](https://doi.org/10.1109/ACCESS.2021.3119562).
- [92] C.-Y. Wang, A. Bochkovskiy, and H.-Y. Mark Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022, *arXiv:2207.02696*.
- [93] L. Zhang, Z. Deng, K. Kawaguchi, A. Ghorbani, and J. Zou, "How does mixup help with robustness and generalization?" 2020, *arXiv:2010.04819*.
- [94] G. Ghiasi, Y. Cui, A. Srinivas, R. Qian, T.-Y. Lin, E. D. Cubuk, Q. V. Le, and B. Zoph, "Simple copy-paste is a strong data augmentation method for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 2917–2927, doi: [10.1109/CVPR46437.2021.00294](https://doi.org/10.1109/CVPR46437.2021.00294).
- [95] M. Mahasin and I. A. Dewi, "Comparison of CSPDarkNet53, CSPResNeXt-50, and EfficientNet-B0 backbones on YOLO V4 as object detector," *Int. J. Eng., Sci. Inf. Technol.*, vol. 2, no. 3, pp. 64–72, Sep. 2022, doi: [10.52088/ijesty.v2i3.291](https://doi.org/10.52088/ijesty.v2i3.291).
- [96] K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng, "PANet: Few-shot image semantic segmentation with prototype alignment," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9196–9205, doi: [10.1109/ICCV.2019.00929](https://doi.org/10.1109/ICCV.2019.00929).
- [97] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv9: Learning what you want to learn using programmable gradient information," 2024, *arXiv:2402.13616*.
- [98] L. Wang, S. Letchmunan, and R. Xiao, "Gelan-SE: Squeeze and stimulus attention based target detection network for gelan architecture," *IEEE Access*, vol. 12, pp. 182259–182273, 2024, doi: [10.1109/ACCESS.2024.3462725](https://doi.org/10.1109/ACCESS.2024.3462725).
- [99] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding, "YOLOv10: Real-time end-to-end object detection," 2024, *arXiv:2405.14458*.
- [100] R. Khanam and M. Hussain, "YOLOv11: An overview of the key architectural enhancements," 2024, *arXiv:2410.17725*.
- [101] R. Sapkota, R. Qureshi, M. F. Calero, C. Badjagar, U. Nepal, A. Poullose, P. Zeno, U. B. P. Vaddevolu, S. Khan, M. Shoman, H. Yan, and M. Karkee, "YOLOv12 to its genesis: A decadal and comprehensive review of the you only look once (YOLO) series," 2024, *arXiv:2406.19407*.
- [102] A. F. Rasheed and M. Zarkoosh, "YOLOv11 optimization for efficient resource utilization," 2024, *arXiv:2412.14790*.
- [103] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-centric real-time object detectors," 2025, *arXiv:2502.12524*.
- [104] M. A. R. Alif and M. Hussain, "YOLOv12: A breakdown of the key architectural features," 2025, *arXiv:2502.14740*.
- [105] T. Dao, "FlashAttention-2: Faster attention with better parallelism and work partitioning," 2023, *arXiv:2307.08691*.
- [106] J. Shah, G. Bikshandi, Y. Zhang, V. Thakkar, P. Ramani, and T. Dao, "FlashAttention-3: Fast and accurate attention with asynchrony and low-precision," 2024, *arXiv:2407.08608*.
- [107] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré, "FlashAttention: Fast and memory-efficient exact attention with IO-awareness," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 17041–17055.
- [108] A. S. Geetha, "YOLOv4: A breakthrough in real-time object detection," 2025, *arXiv:2502.04161*.
- [109] M. Hussain, "YOLOv5, YOLOv8 and YOLOv10: The go-to detectors for real-time vision," 2024, *arXiv:2407.02988*.
- [110] A. S. Geetha, "What is YOLOv6? A deep insight into the object detection model," 2024, *arXiv:2412.13006*.
- [111] G. Jocher. (2023). *YOLOv8*. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [112] M. Goin and J. Finks. (Jan. 18, 2023). *YOLOv8 Detection 10x Faster With DeepSparse-Over 500 FPS on a CPU*. Neural Magic Blog. Accessed: Mar. 12, 2025. [Online]. Available: <https://neuralmagic.com/blog/yolov8-detection-10x-faster-with-deepsparse-500-fps-on-a-cpu/>
- [113] G. Jocher. *YOLOv9: A Leap Forward in Object Detection Technology*. Ultralytics GitHub Repository. Accessed: Mar. 12, 2025. [Online]. Available: <https://github.com/ultralytics/ultralytics/blob/main/docs/en/models/yolov9.md>
- [114] P. Hidayatullah, N. Syakrani, M. R. Sholahuddin, T. Gelar, and R. Tubagus, "YOLOv8 to YOLOv11: A comprehensive architecture in-depth comparative review," 2025, *arXiv:2501.13400*.
- [115] X. Zhang, H. Wang, and H. Shi, "Light-weight YOLOv5 for real-time object detection in remote sensing images on embedded artificial intelligence chips," in *Proc. 5th Int. Conf. Artif. Intell. Electromech. Autom. (AIEA)*, Jun. 2024, doi: [10.1109/aiea62095.2024.10692971](https://doi.org/10.1109/aiea62095.2024.10692971).
- [116] G. Jocher. *YOLOv5 Release V7.0*. Accessed: Mar. 13, 2025. [Online]. Available: <https://github.com/ultralytics/yolov5/releases/tag/v7.0>
- [117] C. Li, L. Li, Y. Geng, H. Jiang, M. Cheng, B. Zhang, Z. Ke, X. Xu, and X. Chu, "YOLOv6 v3.0: A full-scale reloading," 2023, *arXiv:2301.05586*.
- [118] X. Long, K. Deng, G. Wang, Y. Zhang, Q. Dang, Y. Gao, H. Shen, J. Ren, S. Han, E. Ding, and S. Wen, "PP-YOLO: An effective and efficient implementation of object detector," 2020, *arXiv:2007.12099*.
- [119] I. Ashraf, S. Hur, G. Kim, and Y. Park, "Analyzing performance of YOLOx for detecting vehicles in bad weather conditions," *Sensors*, vol. 24, no. 2, p. 522, Jan. 2024, doi: [10.3390/s24020522](https://doi.org/10.3390/s24020522).
- [120] H. Wei, Q. Zhang, Y. Qin, X. Li, and Y. Qian, "YOLOF-F: You only look one-level feature fusion for traffic sign detection," *Vis. Comput.*, vol. 40, no. 2, pp. 747–760, Feb. 2024, doi: [10.1007/s00371-023-02813-1](https://doi.org/10.1007/s00371-023-02813-1).
- [121] Z. Liu, G. Liao, L. Cao, and Y. Yang, "YOLOv5-lite: A novel compression method based on channel pruning," *Eng. Res. Exp.*, vol. 7, no. 2, Jun. 2025, Art. no. 025231, doi: [10.1088/2631-8695/adcff8](https://doi.org/10.1088/2631-8695/adcff8).
- [122] Y. A. Marvell, A. A. S. Gunawan, and I. A. Iswanto, "Enhancing small object detection: Development on car logo identification using the YOLO-NAS method for speed and accuracy improvements," *Proc. Comput. Sci.*, vol. 245, pp. 528–537, May 2024, doi: [10.1016/j.procs.2024.10.279](https://doi.org/10.1016/j.procs.2024.10.279).
- [123] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "YOLO-world: Real-time open-vocabulary object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2024, pp. 16901–16911, doi: [10.1109/CVPR52733.2024.01599](https://doi.org/10.1109/CVPR52733.2024.01599).

- [124] L. Liu, J. Feng, H. Chen, A. Wang, L. Song, J. Han, and G. Ding, "YOLO-UniOW: Efficient universal open-world object detection," 2024, *arXiv:2412.20645*.
- [125] A. Abane, M. Cubeddu, V. Sy Mai, and A. Battou, "Entanglement routing in quantum networks: A comprehensive survey," 2024, *arXiv:2408.01234*.
- [126] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. 18th Int. Conf. Pattern Recognit. (ICPR)*, 2006, pp. 850–855, doi: [10.1109/icpr.2006.479](#).
- [127] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proc. AAAI*, Apr. 2020, vol. 34, no. 7, pp. 12993–13000, doi: [10.1609/aaai.v34i07.6999](#).
- [128] B. Cheng, R. Girshick, P. Dollár, A. C. Berg, and A. Kirillov, "Boundary IoU: Improving object-centric image segmentation evaluation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 15329–15337, doi: [10.1109/CVPR46437.2021.01508](#).
- [129] Y.-J. Chang, Y.-D. Wu, Y.-L. Liao, C.-H. Wang, and Y.-C. Wu, "Modified YOLO network model for metaphase cell detection in antinuclear antibody images," *Eng. Appl. Artif. Intell.*, vol. 127, Jan. 2024, Art. no. 107392, doi: [10.1016/j.engappai.2023.107392](#).
- [130] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
- [131] E. Lee, J.-S. Kim, D. K. Park, and T. Whangbo, "YOLO-MR: Meta-learning-based lesion detection algorithm for resolving data imbalance," *IEEE Access*, vol. 12, pp. 49762–49771, 2024, doi: [10.1109/ACCESS.2024.3384088](#).
- [132] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, Jun. 2017, pp. 1–13.
- [133] Q. Fan, Y. Li, M. Deveci, K. Zhong, and S. Kadry, "LUD-YOLO: A novel lightweight object detection network for unmanned aerial vehicle," *Inf. Sci.*, vol. 686, Jan. 2025, Art. no. 121366, doi: [10.1016/j.ins.2024.121366](#).
- [134] D. Du et al., "VisDrone-DET2019: The vision meets drone object detection in image challenge results," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 213–226.
- [135] P. Zhu et al., "VisDrone-VID2019: The vision meets drone object detection in video challenge results," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 227–235, doi: [10.1109/ICCVW.2019.00031](#).
- [136] X. Wei, Z. Li, and Y. Wang, "SED-YOLO based multi-scale attention for small object detection in remote sensing," *Sci. Rep.*, vol. 15, no. 1, p. 3125, Jan. 2025, doi: [10.1038/s41598-025-87199-x](#).
- [137] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3974–3983, doi: [10.1109/CVPR.2018.00418](#).
- [138] Y. Tian, S. Wang, E. Li, G. Yang, Z. Liang, and M. Tan, "MD-YOLO: Multi-scale dense YOLO for small target pest detection," *Comput. Electron. Agricult.*, vol. 213, Oct. 2023, Art. no. 108233, doi: [10.1016/j.compag.2023.108233](#).
- [139] Y. Zhu and S. Newsam, "DenseNet for dense flow," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 790–794, doi: [10.1109/ICIP.2017.8296389](#).
- [140] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, and K. Keutzer, "DenseNet: Implementing efficient ConvNet descriptor pyramids," 2014, *arXiv:1404.1869*.
- [141] L. Wang, Y. Cao, S. Wang, X. Song, S. Zhang, J. Zhang, and J. Niu, "Investigation into recognition algorithm of helmet violation based on YOLOv5-CBAM-DCN," *IEEE Access*, vol. 10, pp. 60622–60632, 2022, doi: [10.1109/ACCESS.2022.3180796](#).
- [142] F. Chen, F. Wu, J. Xu, G. Gao, Q. Ge, and X.-Y. Jing, "Adaptive deformable convolutional network," *Neurocomputing*, vol. 453, pp. 853–864, Sep. 2021, doi: [10.1016/j.neucom.2020.06.128](#).
- [143] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Computer Vision—ECCV*. Cham, Switzerland: Springer, 2018, pp. 3–19, doi: [10.1007/978-3-030-01234-2_1](#).
- [144] L. Cheng, "A highly robust helmet detection algorithm based on YOLO V8 and transformer," *IEEE Access*, vol. 12, pp. 130693–130705, 2024, doi: [10.1109/ACCESS.2024.3459591](#).
- [145] M. Soudy, Y. Afify, and N. Badr, "RepConv: A novel architecture for image scene classification on Intel scenes dataset," *Int. J. Intell. Comput. Inf. Sci.*, vol. 22, no. 2, pp. 63–73, May 2022, doi: [10.21608/ijci.2022.118834.1163](#).
- [146] M. Kaushal, "Rapid -YOLO: A novel YOLO based architecture for shadow detection," *Optik*, vol. 260, Jun. 2022, Art. no. 169084, doi: [10.1016/j.jleo.2022.169084](#).
- [147] G. Kaur and S. Arora, "Chaotic whale optimization algorithm," *J. Comput. Design Eng.*, vol. 5, no. 3, pp. 275–284, Jul. 2018, doi: [10.1016/j.jcde.2017.12.006](#).
- [148] Y. Li, S. Li, H. Du, L. Chen, D. Zhang, and Y. Li, "YOLO-ACN: Focusing on small target and occluded object detection," *IEEE Access*, vol. 8, pp. 227288–227303, 2020, doi: [10.1109/ACCESS.2020.3046515](#).
- [149] S. X. Tan, J. Y. Ong, K. O. M. Goh, and C. Tee, "Boosting vehicle classification with augmentation techniques across multiple YOLO versions," *JOIV, Int. J. Informat. Visualizat.*, vol. 8, no. 1, p. 45, Mar. 2024, doi: [10.62527/joiv.8.1.2313](#).
- [150] C. Niu, Y. Song, and X. Zhao, "SE-lightweight YOLO: Higher accuracy in YOLO detection for vehicle inspection," *Appl. Sci.*, vol. 13, no. 24, p. 13052, Dec. 2023, doi: [10.3390/app132413052](#).
- [151] Z. Guo, C. Wang, G. Yang, Z. Huang, and G. Li, "MSFT-YOLO: Improved YOLOv5 based on transformer for detecting defects of steel surface," *Sensors*, vol. 22, no. 9, p. 3467, May 2022, doi: [10.3390/s22093467](#).
- [152] Z. Zhang, Q. Tong, and X. Huang, "An efficient YOLO network with CSPCBAM, ghost, and cluster-NMS for underwater target detection," *IEEE Access*, vol. 12, pp. 30562–30576, 2024, doi: [10.1109/ACCESS.2024.3368878](#).
- [153] N. Zaghari, M. Fathy, S. M. Jameii, and M. Shahverdy, "The improvement in obstacle detection in autonomous vehicles using YOLO non-maximum suppression fuzzy algorithm," *J. Supercomput.*, vol. 77, no. 11, pp. 13421–13446, Nov. 2021, doi: [10.1007/s11227-021-03813-5](#).
- [154] N. Aishwarya, G. S. Yayathish Kanna, and K. Seemakurthy, "YOLOskin: A fusion framework for improved skin cancer diagnosis using YOLO detectors on Nvidia Jetson nano," *Biomed. Signal Process. Control*, vol. 100, Feb. 2025, Art. no. 107093, doi: [10.1016/j.bspc.2024.107093](#).
- [155] G. C. Sunil, A. Upadhyay, Y. Zhang, K. Howatt, T. Peters, M. Ostlie, W. Aderholdt, and X. Sun, "Field-based multispecies weed and crop detection using ground robots and advanced YOLO models: A data and model-centric approach," *Smart Agricult. Technol.*, vol. 9, Dec. 2024, Art. no. 100538, doi: [10.1016/j.atech.2024.100538](#).
- [156] N. Rane, "YOLO and faster R-CNN object detection for smart industry 4.0 and industry 5.0: Applications, challenges, and opportunities," *SSRN Electron. J.*, Nov. 2023, doi: [10.2139/ssrn.4624206](#).
- [157] J. E. Gallagher and E. J. Oughton, "Surveying you only look once (YOLO) multispectral object detection advancements, applications, and challenges," *IEEE Access*, vol. 13, pp. 7366–7395, 2025, doi: [10.1109/ACCESS.2025.3526458](#).
- [158] H. Wu, X. Mo, S. Wen, K. Wu, Y. Ye, Y. Wang, and Y. Zhang, "DNE-YOLO: A method for apple fruit detection in diverse natural environments," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 36, no. 9, Nov. 2024, Art. no. 102220, doi: [10.1016/j.jksuci.2024.102220](#).
- [159] N. Azatbekuly, A. Mukhanbet, and S. D. Bekele, "Development of an intelligent video surveillance system based on YOLO algorithm," in *Proc. IEEE 4th Int. Conf. Smart Inf. Syst. Technol. (SIST)*, Astana, Kazakhstan, May 2024, pp. 498–503, doi: [10.1109/SIST61555.2024.10629617](#).
- [160] M. Almelhdhar, A. Albaser, M. A. Khan, M. Abdallah, H. Menouar, S. Al-Kuwari, and A. Al-Fuqaha, "Deep learning in the fast lane: A survey on advanced intrusion detection systems for intelligent vehicle networks," *IEEE Open J. Veh. Technol.*, vol. 5, pp. 869–906, 2024, doi: [10.1109/OJVT.2024.3422253](#).
- [161] S. Chen and W. Lin, "Embedded system real-time vehicle detection based on improved YOLO network," in *Proc. IEEE 3rd Adv. Inf. Manage., Communicates, Electron. Autom. Control Conf. (IMCEC)*, Oct. 2019, pp. 1400–1403.
- [162] S. S. Jamiya and P. E. Rani, "LittleYOLO-SPP: A delicate real-time vehicle detection algorithm," *Optik*, vol. 225, Jan. 2021, Art. no. 165818, doi: [10.1016/j.jleo.2020.165818](#).
- [163] T. O. Hodson, T. M. Over, and S. S. Foks, "Mean squared error, deconstructed," *J. Adv. Model. Earth Syst.*, vol. 13, no. 12, p. 2021, Dec. 2021, doi: [10.1029/2021ms002681](#).

- [164] S. Lattanzi and C. Sohler, "A better k-means++ algorithm via local search," in *Proc. 36th Int. Conf. Mach. Learn. (ICML)*, May 2019, pp. 3662–3671.
- [165] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii, "Scalable k-means++," *Proc. VLDB Endowment*, vol. 5, no. 7, pp. 622–633, Mar. 2012, doi: [10.14778/2180912.2180915](#).
- [166] S. Liang, H. Wu, L. Zhen, Q. Hua, S. Garg, G. Kaddoum, M. M. Hassan, and K. Yu, "Edge YOLO: Real-time intelligent object detection system based on edge-cloud cooperation in autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 25345–25360, Dec. 2022, doi: [10.1109/TITS.2022.3158253](#).
- [167] Z. Li, Q. Li, Y. Zhou, W. Zhong, G. Zhang, and C. Wu, "Edge-cloud collaborative learning with federated and centralized features," in *Proc. 46th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2023, pp. 1949–1953, doi: [10.1145/3539618.3591976](#).
- [168] J. Yao, S. Zhang, Y. Yao, F. Wang, J. Ma, J. Zhang, Y. Chu, L. Ji, K. Jia, T. Shen, A. Wu, F. Zhang, Z. Tan, K. Kuang, C. Wu, F. Wu, J. Zhou, and H. Yang, "Edge-cloud polarization and collaboration: A comprehensive survey for AI," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 7, pp. 6866–6886, Jul. 2023, doi: [10.1109/TKDE.2022.3178211](#).
- [169] X. Liu, Y. Wang, D. Yu, and Z. Yuan, "YOLOv8-FDD: A real-time vehicle detection method based on improved YOLOv8," *IEEE Access*, vol. 12, pp. 136280–136296, 2024, doi: [10.1109/ACCESS.2024.3453298](#).
- [170] H. Wei, X. Liu, S. Xu, Z. Dai, Y. Dai, and X. Xu, "DWRSeg: Rethinking efficient acquisition of multi-scale contextual information for real-time semantic segmentation," 2022, *arXiv:2212.01173*.
- [171] W. Liu, H. Lu, H. Fu, and Z. Cao, "Learning to upsample by learning to sample," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 6004–6014.
- [172] S. Lei, H. Yi, and J. S. Sarmiento, "Synchronous end-to-end vehicle pedestrian detection algorithm based on improved YOLOv8 in complex scenarios," *Sensors*, vol. 24, no. 18, p. 6116, Sep. 2024, doi: [10.3390/s24186116](#).
- [173] Y. Zhang, Y. Sun, Z. Wang, and Y. Jiang, "YOLOv7-RAR for urban vehicle detection," *Sensors*, vol. 23, no. 4, p. 1801, Feb. 2023, doi: [10.3390/s23041801](#).
- [174] L. Wen, D. Du, Z. Cai, Z. Lei, M.-C. Chang, H. Qi, J. Lim, M.-H. Yang, and S. Lyu, "UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking," *Comput. Vis. Image Understand.*, vol. 193, Apr. 2020, Art. no. 102907, doi: [10.1016/j.cviu.2020.102907](#).
- [175] L. Xu, W. Yan, and J. Ji, "The research of a novel WOG-YOLO algorithm for autonomous driving object detection," *Sci. Rep.*, vol. 13, no. 1, p. 3699, Mar. 6, 2023, doi: [10.1038/s41598-023-30409-1](#).
- [176] H. Xu, M. Guo, N. Nedjah, J. Zhang, and P. Li, "Vehicle and pedestrian detection algorithm based on lightweight YOLOv3-promote and semi-precision acceleration," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 19760–19771, Oct. 2022, doi: [10.1109/TITS.2021.3137253](#).
- [177] T. Zhang, W. Xu, B. Luo, and G. Wang, "Depth-wise convolutions in vision transformers for efficient training on small datasets," *Neurocomputing*, vol. 617, Feb. 2025, Art. no. 128998, doi: [10.1016/j.neucom.2024.128998](#).
- [178] S.-J. Ji, Q.-H. Ling, and F. Han, "An improved algorithm for small object detection based on YOLO v4 and multi-scale contextual information," *Comput. Electr. Eng.*, vol. 105, Jan. 2023, Art. no. 108490, doi: [10.1016/j.compeleceng.2022.108490](#).
- [179] J. Tang, H. Lai, G. Gao, and T. Wang, "PFEL-Net: A lightweight network to enhance feature for multi-scale pedestrian detection," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 36, no. 8, p. 102198, Oct. 2024, doi: [10.1016/j.jksuci.2024.102198](#).
- [180] W. Lan, J. Dang, Y. Wang, and S. Wang, "Pedestrian detection based on YOLO network model," in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Changchun, China, Aug. 2018, pp. 1547–1551.
- [181] Q. Liu, Z. Li, L. Zhang, and J. Deng, "MSCD-YOLO: A lightweight dense pedestrian detection model with finer-grained feature information interaction," *Sensors*, vol. 25, no. 2, p. 438, Jan. 2025, doi: [10.3390/s25020438](#).
- [182] Y. Xue, Z. Ju, Y. Li, and W. Zhang, "MAF-YOLO: Multi-modal attention fusion based YOLO for pedestrian detection," *Infr. Phys. Technol.*, vol. 118, Nov. 2021, Art. no. 103906, doi: [10.1016/j.infrared.2021.103906](#).
- [183] H. Zhu, X. Chen, W. Dai, K. Fu, Q. Ye, and J. Jiao, "Orientation robust object detection in aerial images using deep convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, QC, Canada, Sep. 2015, pp. 3735–3739, doi: [10.1109/ICIP.2015.7351502](#).
- [184] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, and J. Sun, "CrowdHuman: A benchmark for detecting human in a crowd," 2018, *arXiv:1805.00123*.
- [185] Y. Choi, N. Kim, S. Hwang, K. Park, J. S. Yoon, K. An, and I. S. Kweon, "KAIST multi-spectral day/night data set for autonomous and assisted driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 934–948, Mar. 2018, doi: [10.1109/TITS.2018.2791533](#).
- [186] R. Zhao, S. H. Tang, J. Shen, E. E. B. Supeni, and S. A. Rahim, "Enhancing autonomous driving safety: A robust traffic sign detection and recognition model TSD-YOLO," *Signal Process.*, vol. 225, Dec. 2024, Art. no. 109619, doi: [10.1016/j.sigpro.2024.109619](#).
- [187] Y. Sun, X. Li, D. Zhao, and Q.-G. Wang, "Evolving traffic sign detection via multi-scale feature enhancement, reconstruction and fusion," *Digit. Signal Process.*, vol. 160, May 2025, Art. no. 105028, doi: [10.1016/j.dsp.2025.105028](#).
- [188] Y. Han, F. Wang, W. Wang, X. Zhang, and X. Li, "EDN-YOLO: Multi-scale traffic sign detection method in complex scenes," *Digit. Signal Process.*, vol. 153, Oct. 2024, Art. no. 104615, doi: [10.1016/j.dsp.2024.104615](#).
- [189] B. Meng and W. Shi, "Small traffic sign recognition method based on improved YOLOv7," *Sci. Rep.*, vol. 15, no. 1, p. 5482, Feb. 2025, doi: [10.1038/s41598-025-88679-w](#).
- [190] M. Niu, Y. Chen, J. Li, X. Qiu, and W. Cai, "YOLOv8s-DDA: An improved small traffic sign detection algorithm based on YOLOv8s," *Electronics*, vol. 13, no. 18, p. 3764, Sep. 2024, doi: [10.3390/electronics13183764](#).
- [191] J. Luo and Z. Wang, "A low latency traffic sign detection model with an automatic data labeling pipeline," *Neural Comput. Appl.*, vol. 34, no. 18, pp. 15499–15512, Sep. 2022.
- [192] S. Garg, A. Baghel, A. Agarwal, and D. Toshniwal, "Snowy scenes, clear detections: A robust model for traffic light detection in adverse weather conditions," 2024, *arXiv:2406.13473*.
- [193] M. Bakirci, "Utilizing YOLOv8 for enhanced traffic monitoring in intelligent transportation systems (ITS) applications," *Digit. Signal Process.*, vol. 152, Sep. 2024, Art. no. 104594, doi: [10.1016/j.dsp.2024.104594](#).
- [194] J. Choi and H. Lee, "Real-time traffic light recognition with lightweight state recognition and ratio-preserving zero padding," *Electronics*, vol. 13, no. 3, p. 615, Feb. 2024, doi: [10.3390/electronics13030615](#).
- [195] F. Wei and W. Wang, "SCCA-YOLO: A spatial and channel collaborative attention enhanced YOLO network for highway autonomous driving perception system," *Sci. Rep.*, vol. 15, no. 1, p. 6459, Feb. 2025, doi: [10.1038/s41598-025-90743-4](#).
- [196] H. T. Ngoc, K. H. Nguyen, H. K. Hua, H. V. N. Nguyen, and L.-D. Quach, "Optimizing YOLO performance for traffic light detection and end-to-end steering control for autonomous vehicles in gazebo-ROS2," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 7, pp. 475–484, 2023, doi: [10.14569/IJACSA.2023.0140752](#).
- [197] B. Liu, H. Wang, Y. Wang, C. Zhou, and L. Cai, "Lane line type recognition based on improved YOLOv5," *Appl. Sci.*, vol. 13, no. 18, p. 10537, Sep. 2023, doi: [10.3390/app131810537](#).
- [198] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 11531–11539, doi: [10.1109/CVPR42600.2020.01155](#).
- [199] X. Zhang, W. Yang, X. Tang, and J. Liu, "A fast learning method for accurate and robust lane detection using two-stage feature extraction with YOLO v3," *Sensors*, vol. 18, no. 12, p. 4308, Dec. 2018, doi: [10.3390/s18124308](#).
- [200] L. Ding and A. Goshtasby, "On the Canny edge detector," *Pattern Recognit.*, vol. 34, no. 3, pp. 721–725, Mar. 2001, doi: [10.1016/S0031-3203\(00\)00023-6](#).
- [201] S. Shirke and R. Udayakumar, "Lane datasets for lane detection," in *Proc. Int. Conf. Commun. Signal Process. (ICCCSP)*, Chennai, India, Apr. 2019, pp. 0792–0796, doi: [10.1109/ICCCSP.2019.8698065](#).
- [202] D. Li, Z. Yang, W. Nai, Y. Xing, and Z. Chen, "A road lane detection approach based on reformer model," *Egyptian Informat. J.*, vol. 29, Mar. 2025, Art. no. 100625, doi: [10.1016/j.eij.2025.100625](#).
- [203] N. Kitaev, Ł. Kaiser, and A. Levskaya, "Reformer: The efficient transformer," 2020, *arXiv:2001.04451*.

- [204] M. Kaviani, J. Almeida, and F. L. Verdi, "Residual-based adaptive Huber loss (RAHL)—Design of an improved Huber loss for CQI prediction in 5G networks," in *Proc. Anais do 42nd Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, Niterói, Brazil, May 2024, pp. 602–615, doi: [10.5753/sbrc.2024.1446](https://doi.org/10.5753/sbrc.2024.1446).
- [205] L. Wang, C. Zheng, W. Zhou, and W.-X. Zhou, "A new principle for tuning-free Huber regression," *Stat Sinica*, vol. 31, no. 4, pp. 2153–2177, 2021, doi: [10.5705/ss.202019.0045](https://doi.org/10.5705/ss.202019.0045).
- [206] P. N. Huu, Q. P. Thi, and P. T. T. Quynh, "Proposing lane and obstacle detection algorithm using YOLO to control self-driving cars on advanced networks," *Adv. Multimedia*, vol. 2022, pp. 1–18, May 2022, doi: [10.1155/2022/3425295](https://doi.org/10.1155/2022/3425295).
- [207] Z. Fu, W. Song, Y. Wang, X. Wu, Y. Zheng, Y. Zhang, D. Xu, X. Wei, T. Xu, and X. Zhao, "Sliding window attention training for efficient large language models," 2025, *arXiv:2502.18845*.
- [208] T. Deng and Y. Wu, "Simultaneous vehicle and lane detection via MobileNetV3 in car following scene," *PLoS ONE*, vol. 17, no. 3, Mar. 2022, Art. no. e0264551, doi: [10.1371/journal.pone.0264551](https://doi.org/10.1371/journal.pone.0264551).
- [209] L. Q. Zhao and L. L. Wang, "A new lightweight network based on MobileNetV3," *KSII Trans. Internet Inf. Syst.*, vol. 16, pp. 1–15, Jan. 2022, doi: [10.3837/tiis.2022.01.001](https://doi.org/10.3837/tiis.2022.01.001).
- [210] V. Badrinarayanan, A. Handa, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling," 2015, *arXiv:1505.07293*.
- [211] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, "Spatial as deep: Spatial CNN for traffic scene understanding," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, Apr. 2018, pp. 7276–7283, doi: [10.1609/aaai.v32i1.12301](https://doi.org/10.1609/aaai.v32i1.12301).
- [212] T. Sato and Q. A. Chen, "Towards driving-oriented metric for lane detection models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 17132–17141, doi: [10.1109/CVPR52688.2022.01664](https://doi.org/10.1109/CVPR52688.2022.01664).
- [213] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3D object detection methods for autonomous driving applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3782–3795, Oct. 2019.
- [214] M. M. Islam, A. A. R. Newaz, and A. Karimoddini, "A pedestrian detection and tracking framework for autonomous cars: Efficient fusion of camera and LiDAR data," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2021, pp. 1287–1292, doi: [10.1109/SMC52423.2021.9658639](https://doi.org/10.1109/SMC52423.2021.9658639).
- [215] P. Shi, Z. Liu, X. Dong, and A. Yang, "CL-fusionBEV: 3D object detection method with camera-LiDAR fusion in Bird's eye view," *Complex Intell. Syst.*, vol. 10, no. 6, pp. 7681–7696, Dec. 2024, doi: [10.1007/s40747-024-01567-0](https://doi.org/10.1007/s40747-024-01567-0).
- [216] Y. Jin, X. Zhu, Y. Yue, and J. Ma, "A hybrid model for object detection based on feature-level camera-radar fusion in autonomous driving," in *Proc. 8th Int. Conf. Intell. Comput. Signal Process. (ICSP)*, Xi'an, China, Apr. 2023, pp. 897–903, doi: [10.1109/ICSP58490.2023.10248746](https://doi.org/10.1109/ICSP58490.2023.10248746).
- [217] Z. A. T. Rakotoniaina, N. E. Chelbi, D. Gingras, and F. Faulconier, "LiDAR and radar sensor fusion for detection and tracking of dynamic objects in autonomous vehicles using probabilistic occupancy grid, YOLO and DeepSORT," in *Proc. IEEE 25th Int. Conf. Intell. Transp. Syst. (ITSC)*, Macau, China, Oct. 2022, pp. 4131–4137, doi: [10.1109/ITSC55140.2022.9922271](https://doi.org/10.1109/ITSC55140.2022.9922271).
- [218] M. Carranza-García, F. J. Galán-Sales, J. M. Luna-Romera, and J. C. Riquelme, "Object detection using depth completion and camera-LiDAR fusion for autonomous driving," *Integr. Comput.-Aided Eng.*, vol. 29, no. 3, pp. 241–258, Jun. 2022, doi: [10.3233/ica-220681](https://doi.org/10.3233/ica-220681).
- [219] R. Murendeni, A. Mwanza, and I. C. Obagbuwa, "Using a YOLO deep learning algorithm to improve the accuracy of 3D object detection by autonomous vehicles," *World Electr. Vehicle J.*, vol. 16, no. 1, p. 9, Dec. 2024, doi: [10.3390/wevj16010009](https://doi.org/10.3390/wevj16010009).
- [220] X. Nie, L. Zhu, Z. He, A. Cheng, S. Zhong, and E. Li, "Investigating 3D object detection using stereo camera and LiDAR fusion with bird's-eye view representation," *Neurocomputing*, vol. 620, Mar. 2025, Art. no. 129144.
- [221] S. Nordhoff, J. D. Lee, S. C. Calvert, S. Berge, M. Hagenzieker, and R. Happee, "(Mis-)use of standard autopilot and full self-driving (FSD) beta: Results from interviews with users of Tesla's FSD beta," *Frontiers Psychol.*, vol. 14, Feb. 2023, Art. no. 1101520, doi: [10.3389/fpsyg.2023.1101520](https://doi.org/10.3389/fpsyg.2023.1101520).
- [222] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. Qi, Y. Zhou, Z. Yang, A. Chouard, P. Sun, J. Ngiam, V. Vasudevan, A. McCauley, J. Shlens, and D. Anguelov, "Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 9690–9699, doi: [10.1109/ICCV48922.2021.00957](https://doi.org/10.1109/ICCV48922.2021.00957).
- [223] Z. Gu, Z. Li, X. Di, and R. Shi, "An LSTM-based autonomous driving model using a waymo open dataset," *Appl. Sci.*, vol. 10, no. 6, p. 2046, Mar. 2020, doi: [10.3390/app10062046](https://doi.org/10.3390/app10062046).
- [224] H. Fan, F. Zhu, C. Liu, L. Zhang, L. Zhuang, D. Li, W. Zhu, J. Hu, H. Li, and Q. Kong, "Baidu Apollo EM motion planner," 2018, *arXiv:1807.08048*.
- [225] K. Xu, X. Xiao, J. Miao, and Q. Luo, "Data driven prediction architecture for autonomous driving and its application on Apollo platform," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2020, pp. 175–181, doi: [10.1109/IV47402.2020.9304810](https://doi.org/10.1109/IV47402.2020.9304810).
- [226] S. Kato, S. Tokunaga, Y. Maruyama, S. Maeda, M. Hirabayashi, Y. Kitsukawa, A. Monrroy, T. Ando, Y. Fujii, and T. Azumi, "Autoware on board: Enabling autonomous vehicles with embedded systems," in *Proc. ACM/IEEE 9th Int. Conf. Cyber-Phys. Syst. (ICCPs)*, Apr. 2018, pp. 287–296, doi: [10.1109/ICCPs.2018.00035](https://doi.org/10.1109/ICCPs.2018.00035).
- [227] H.-Y. Jung, D.-H. Paek, and S.-H. Kong, "Open-source autonomous driving software platforms: Comparison of autoware and Apollo," 2025, *arXiv:2501.18942*.



JIAPPEI WEI received the B.S. and M.S. degrees in vehicle engineering from Lanzhou Jiaotong University, Gansu, China, in 2008 and 2013, respectively. He is currently pursuing the Ph.D. degree with Universiti Putra Malaysia.

He is currently a Lecturer with Tianshui Normal University, Gansu, China. His current research interests include deep learning, object detection, and intelligent vehicle engineering.



AZIZAN AS'ARRY received the bachelor's, M.Sc., and Ph.D. degrees from the Universiti Teknologi Malaysia (UTM), in 2008, 2009, and 2014, respectively.

He is currently a Senior Lecturer with the Department of Mechanical and Manufacturing Engineering, Faculty of Engineering, Universiti Putra Malaysia. He is the Head of the Instrumentation and Control Systems Laboratory within the same department. His major fields of study are intelligence control and mechatronics. His research interests include mechatronics, active vibration control, intelligent optimization, system identification, the IoT automation, and robotics.



KHAIRIL ANAS MD REZALI received the B.Eng. in mechanical engineering from Universiti Putra Malaysia, in 2010, and the Ph.D. degree in acoustics and vibrations from the Institute of Sound and Vibration, University of Southampton, U.K., in 2015.

He is currently a Ph.D. Supervisor with the Department of Mechanical and Manufacturing Engineering, Universiti Putra Malaysia. His research interests include mechanical vibration, biodynamics, human vibration, vibration isolation, and ergonomics.

Dr. Rezali is a Graduate Member of the Board of Engineers Malaysia (BEM), a member of the Society of Automotive Engineers (SAE), a Graduate Member of the Institution of Engineers Malaysia (IEM), and a member of the Society of Vibrations and Acoustics Malaysia (SVAM).



MOHD ZUHRI MOHAMED YUSOFF received the B.Eng. degree (Hons.) in mechanical engineering (design and innovation) from the Universiti Teknikal Malaysia Melaka, the M.Sc. degree in materials and design engineering from the Universiti Putra Malaysia, and the Ph.D. degree in engineering from the University of Liverpool, U.K.

Currently, he is a Senior Lecturer with the Department of Mechanical and Manufacturing Engineering, Universiti Putra Malaysia (UPM), and the Head of the Laboratory of Biocomposite Technology, Institute of Tropical Forestry and Forest Products (INTROP), UPM. To date, he has authored and co-authored over 100 journal articles. His research interests include lightweight composite structures, composite materials (particularly natural fiber composites), and building information modeling (BIM).

Dr. Yusoff is a Graduate Member of the Board of Engineers Malaysia (BEM) and an Active Member of several professional organizations, including the International Association of Advanced Materials (IAAM), the International Association of Engineers (IAENG), the International Society of Mechanical Engineering (ISME), the International Society for Development and Sustainable Development (ISDS), and the Persatuan Pembangunan dan Industri Enau Malaysia (PPIEM).



HAOHAO MA (Graduate Student Member, IEEE) was born in Tianshui, China. He received the bachelor's degree in mechanical design, manufacturing, and automation from the North University of China, in 2011, and the master's degree in mechanical and electronic engineering from Xidian University, China, in 2014. He is currently pursuing the Ph.D. degree with the Universiti Putra Malaysia. He is a Lecturer with Tianshui Normal University, China. His research interests include computer-aided design and manufacturing, robot control, and simulation.



KUNLUN ZHANG received the master's degree in vehicle engineering from Xiamen University of Technology, in 2018. He is currently pursuing the Ph.D. degree with the Universiti Putra Malaysia. Since 2019, he has been with the Institute of Advanced Science and Technology, Beibu Gulf University. His current research interests include artificial intelligence, intelligent control, and automobile vibration reduction.

...