Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.044639

ARTICLE

A New Vehicle Detection Framework Based on Feature-Guided in the Road Scene

Tianmin Deng*, Xiyue Zhang and Xinxin Cheng

School of Traffic and Transportation, Chongqing Jiaotong University, Chongqing, 400074, China

*Corresponding Author: Tianmin Deng. Email: dtianmin@cqjtu.edu.cn

Received: 04 August 2023 Accepted: 13 November 2023 Published: 30 January 2024

ABSTRACT

Vehicle detection plays a crucial role in the field of autonomous driving technology. However, directly applying deep learning-based object detection algorithms to complex road scene images often leads to subpar performance and slow inference speeds in vehicle detection. Achieving a balance between accuracy and detection speed is crucial for real-time object detection in real-world road scenes. This paper proposes a high-precision and fast vehicle detector called the feature-guided bidirectional pyramid network (FBPN). Firstly, to tackle challenges like vehicle occlusion and significant background interference, the efficient feature filtering module (EFFM) is introduced into the deep network, which amplifies the disparities between the features of the vehicle and the background. Secondly, the proposed global attention localization module (GALM) in the model neck effectively perceives the detailed position information of the target, improving both the accuracy and inference speed of the model. Finally, the detection accuracy of small-scale vehicles is further enhanced through the utilization of a four-layer feature pyramid structure. Experimental results show that FBPN achieves an average precision of 60.8% and 97.8% on the BDD100K and KITTI datasets, respectively, with inference speeds reaching 344.83 frames/s and 357.14 frames/s. FBPN demonstrates its effectiveness and superiority by striking a balance between detection accuracy and inference speed, outperforming several state-of-the-art methods.

KEYWORDS

Driverless car; vehicle detection; channel attention mechanism; deep learning

1 Introduction

In recent years, there has been a substantial increase in global vehicle ownership, accompanied by the growing complexity of traffic scenarios, which places significant pressure on driving safety. The adoption of intelligent driving is widely acknowledged as an inevitable solution for transforming transportation systems, given its potential to greatly enhance both safety and efficiency [1]. Generally, intelligent automotive software possesses three primary capabilities: perception, planning, and control [2]. Out of these, the effectiveness of environment perception directly influences the quality of vehicle decision-making and control. Hence, an accurate environment perception system is an essential prerequisite for mitigating traffic accidents [3].

In the past decades, machine vision system [4] has made a substantial and direct impact on a wide range of detection applications. Vision-based vehicle detection [5] has been proven to be a powerful tool with widespread applications in traffic management, intelligent security, and driver assistance systems. The accurate detection of vehicles on urban roads has become indispensable for autonomous driving [6,7]. Correspondently, the core part underpinning accurate detection technology, i.e., vehicle detection algorithm, has been paid significant attention in academic research [8].

Rapid advances in artificial intelligence, particularly in convolutional neural networks (CNNs), have led to significant progress in deep learning, which is now extensively used in object detection [9]. Deep learning-based vehicle detection techniques can generally be classified into one-stage and two-stage methods. In comparison to two-stage algorithms, one-stage algorithms achieve a superior balance between accuracy and speed, making them more suitable for vehicle detection scenarios where speed is of utmost importance [10]. The you only look once (YOLO) family stands out as the most widely adopted detection framework in industrial applications due to its ability to provide a favorable compromise between accuracy and speed. In 2023, Ultralytics launched the YOLOv8 algorithm [11] as an extension of the successful YOLOv5, which includes additional features and improvements. Presently, this updated network outperforms all prior state-of-the-art models for object detection in terms of speed and accuracy.

Despite the advancements in vision-based vehicle detection methods, persistent challenges still remain, for the road traffic environment is complicated and keeps evolving [12,13]. These challenges include variations in lighting conditions, mutual occlusion or overlap among closely parked vehicles, as well as the limited and vulnerable feature information of small target vehicles in the distance, which are easily interfered with by complex backgrounds. Consequently, these factors result in insufficient representation of vehicle details in images, leading to a high ratio of false positives and false negatives in detection. Moreover, in practice, high inference speed is tremendously demanded. Therefore, it is imperative to conduct research on an efficient detection algorithm that will grapple with challenges presented in complex scenes. In scenarios where computational resources are limited and there is a high requirement for accuracy and speed, there remains potential for enhancing the YOLOv8 algorithm. This proposal introduces a novel approach called feature-guided bidirectional pyramid network (FBPN) for vehicle detection in complex environments. It is based on YOLOv8 and consists of two main components: the efficient feature filtering module (EFFM) and the global attention localization module (GALM). The contributions of this work are summarized as follows:

(1) We propose a scheme that strengthens the extraction and fusion process of vehicle feature information from the channel dimension, based on the diversity of information in the feature layers. This scheme achieves both fast inference and high detection accuracy in vehicle detection. The intensive qualitative and quantitative trials conducted on two autonomous driving datasets demonstrate that the designed method achieves a superior balance between speed and accuracy compared to other existing methods.
(2) We design an efficient feature filtering module (EFFM), by leveraging a local channel attention mechanism to generate attention weights among different channels. This mechanism enhances the local feature information for vehicles under complex backgrounds and challenging lighting conditions, including strong light, low light, and backlight. Simultaneously, it effectively suppresses the interference caused by irrelevant occlusion information.
(3) We propose a global attention localization module (GALM) to leverage global contextual information to guide low-level features in accurately selecting the localization detail of detected object categories. Additionally, considering the characteristics of the dataset, the 3-scale detection of the feature detection network is improved to 4-scale detection. As a result, this

enhancement leads to improved detection accuracy for small target vehicles in far-field scenes and reduces the missed detection rate of vehicles.

The remaining part of the paper proceeds as follows: Section 2 provides a concise review of the related literature. Section 3 comprehensively explains the structure of the FBPN algorithm. In Section 4, we present the experimental setup, including the datasets used and the evaluation metrics employed for assessing the performance. In Section 5, experimental results and discussions are presented for two datasets. Finally, the conclusions of this study are presented in Section 6. The code is available at https://github.com/AZKB1122/FBPN.

2 Related Work

Object detection aims to equip autonomous vehicles with the capability to perceive real-world driving scenarios. Due to advancements in computational resources, state-of-the-art object detection algorithms are predominantly based on CNNs. Generally, object detection network models can be categorized into two general groups.

The first category of object detection algorithms is two-stage detection algorithms, which are based on region proposals and demonstrate excellent detection accuracy and precise positioning. This method splits the object detection process into two distinct stages. Firstly, it generates a set of bounding boxes, which is then followed by regression and classification in the subsequent phase. Region with convolution neural network (R-CNN) [14] is the pioneer in the two-stage object detection algorithms. While CNNs have been shown to improve detection accuracy, R-CNN's complex training process and slow detection speed present significant challenges in practical real-life applications. In order to tackle such issues, successive solutions such as Fast R-CNN [15] and Faster R-CNN [16] have been introduced. Notably, Faster R-CNN addresses these challenges by substituting the time-intensive selective search algorithm with a more computationally efficient region proposal network (RPN). Multiple region-based object detection methods have given rise to numerous two-stage techniques specifically designed for vehicle detection. To address the issue of significant scale variations among different vehicle types, Ghosh [17] developed a multi-region proposal network employing multiple RPNs with varying scales to generate regions of interest (ROIs) for detecting and tracking road vehicles. To tackle the challenges of inadequate candidate boxes, suboptimal positive proposal sampling, and underperforming classification in vehicle detection models, Gao et al. [18] employed the fully convolutional one-stage object detection (FCOS) approach in both the RPN and RCNN stages. Their research achieved remarkable enhancements in vehicle detection performance across diverse remote sensing environments. In order to address the challenging task of object detection in adverse conditions, including situations influenced by shadows and varying levels of illumination, a novel model based on Faster R-CNN with optimized neural architecture search (NAS) was proposed [19]. Nonetheless, these approaches encounter challenges related to slow training speed and suboptimal real-time performance, which pose difficulties in meeting the demands of real-time detection.

The second category is called regression-based one-stage detection algorithms. These algorithms simultaneously perform classification and regression for each prior box in the feature map, rendering them highly suitable for real-time applications. The pioneering algorithm in this approach was YOLO [20], followed by the single-shot multi-box detector (SSD) [21]. However, when compared to two-stage object detection algorithms, the YOLO algorithm detects only a limited number of objects from anchors, resulting in lower accuracy. In contrast, SSD enhances the detection accuracy of small objects. Researchers have made improvements to these methods and enhanced object detectors, specifically designed for vehicle detection. For instance, considering the limited computational resources in

autonomous driving systems, Chen et al. [22] used MobileNet v2 to rebuild the backbone network of SSD. Additionally, they incorporated an attention mechanism to enhance the inference speed and precision of their vehicle detection algorithm. To tackle the challenges posed by occlusion in object detection, Deng et al. [23] integrated YOLO with a multi-scale hybrid attention mechanism. Acknowledging the difficulties associated with deploying complex vehicle detection algorithms on mobile devices, they proposed a strategy that utilizes depthwise separable convolution and C3Ghost modules to effectively reduce the model parameters [24].

3 Methods

3.1 Proposed Network Structure

YOLOv8 demonstrates its adaptability to various scenarios by dynamically adjusting the depth and width of the network. Specifically, in this study, we adopted the YOLOv8-S model, with a network width parameter of 0.50 and a depth parameter of 0.33. Building on it, we introduce the FBPN, which enables high-precision real-time detection of vehicles in complex scenarios. The architecture of FBPN is depicted in Fig. 1. To improve the detection performance, EFFM was incorporated into YOLOv8-S. This integration allows for the allocation of more attentional resources to the target area and effectively reduces interference from irrelevant background information. Furthermore, GALM uses the global context generated from high-level features as guidance for low-level features. This module produces merged features that consider both robust semantic information and geometric detailed information. In order to enhance small object detection without compromising the detection performance of other objects, the feature detection network's scale detection has been improved, from 3-scale to 4-scale.

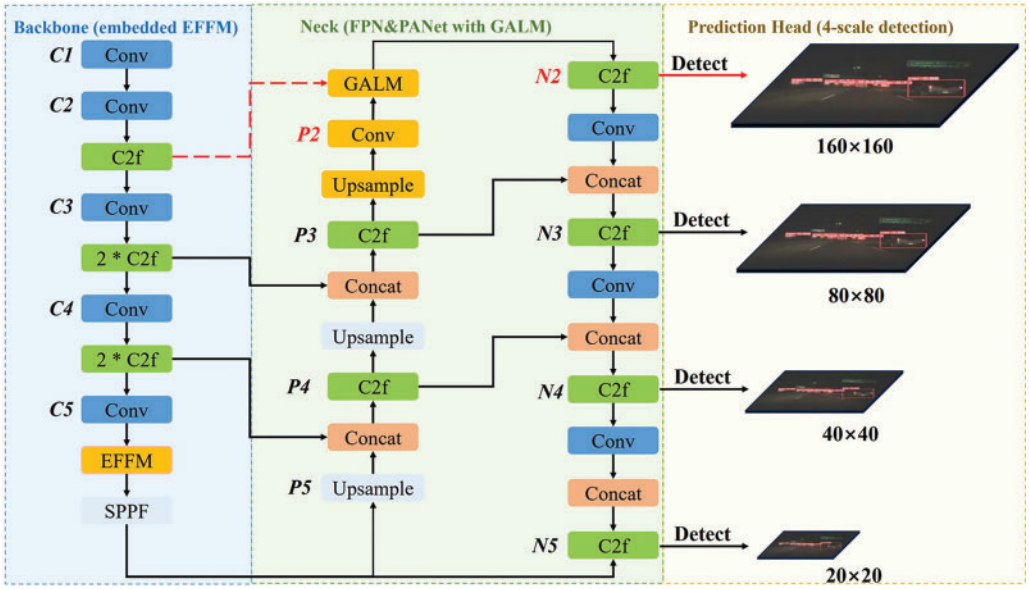

Figure 1: Architecture of the proposed FBPN

3.2 Efficient Feature Filtering Module

In challenging environments, vehicle detection tasks encounter various difficulties. These challenges include complex and unpredictable traffic conditions, diverse lighting conditions (such as backlighting, strong lighting, and low lighting), occlusion between densely arranged vehicles, and

image blurring caused by vehicle motion. All these factors collectively contribute to diminished contrast between the vehicles and backgrounds, thereby impeding the accurate capture of vehicle boundaries. In some circumstances, visual differentiation of the vehicles may be unattainable.

YOLOv8 utilizes C2f modules to achieve a lightweight network and obtain much richer gradient flow information. However, limitations still exist in terms of feature extraction efficiency for vehicle detection in complex environments. This is because C2f modules tend to overlook the significant differences between targets and background feature information during the generation of high-level features. As a result, the feature extraction ability of the original backbone network will be insufficient to meet the requirements of vehicle detection in complex scenarios.

Recently, attention mechanisms have gained significant popularity in enhancing the accuracy of object detection in complex backgrounds [25–27]. The attention mechanisms enable the weighting of feature information based on learned attention weights. Information deemed less relevant is assigned lower weights, whereas information deemed more relevant is assigned higher weights. This allows for the separation of important features from background noise. The spatial attention mechanism is one of the attention mechanisms that is commonly used to assign weights to input feature information from the spatial dimension using operations such as convolution and pooling. However, studies demonstrate that incorporating spatial attention mechanisms does not improve detection performance obviously when applied in scenarios where the target and background characteristics exhibit similarity, making feature extraction especially arduous.

In contrast, channel attention mechanisms have shown superiority in improving feature extraction, which would be a promising alternative to spatial attention mechanisms. These mechanisms assign weights to input feature channels based on their mutual relationships. They assign larger weights to channels containing target features and smaller weights to channels highlighting background noise. ECA-Net [28], for example, enhances the ability to capture inter-channel dependencies by improving the dimensionality reduction operation of SENet [29]. As a lightweight local inter-channel interaction attention module, ECA-Net avoids the massive computational requirements needed to analyze the correlation of each channel to all channels within the global channel scope.

EFFM references ECA-Net and uses local channel attention. The structure of EFFM is shown in Fig. 2.

The local channel attention compresses the input feature map from its spatial dimension to a $1 \times 1 \times C$ feature map through a global average pooling operation. Subsequently, cross-channel interaction information is captured through a one-dimensional convolution. The kernel size, denoted as k, represents the number of neighboring channels involved in the attention prediction for each channel. A nonlinear mapping relationship exists between k and the channel dimension C, which can be expressed mathematically by [28]:

$$k = \left| \frac{log_2 C + b}{\gamma} \right|_{odd} \tag{1}$$

where $|x|_{odd}$ is the nearest odd number of x, C is the channel dimension of the input feature map. In this study, parameters b and γ are assigned values of 1 and 2 throughout all the experiments, respectively.

The weight coefficient for each channel is determined using the sigmoid activation function. This ensures that the weight coefficient falls within the range of 0 to 1, representing the importance of each channel. Ultimately, EFFM assigns these weight coefficients to the input feature space, thereby improving the representation of target features and reducing the impact of background noise.

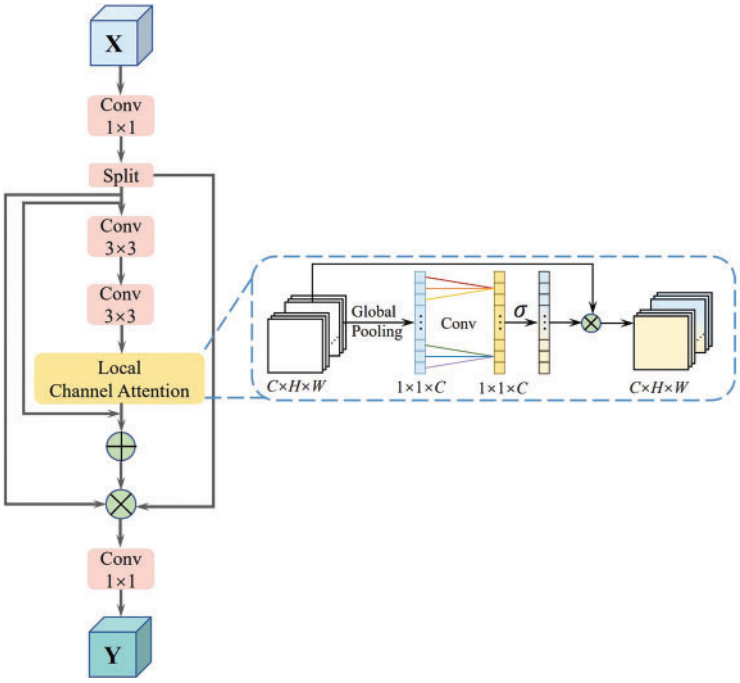

Figure 2: Structure of the EFFM

For its backbone network, YOLOv8 uses cross-stage partial network (CSPNet) as the fundamental concept, which employs five convolutional layers to extract multi-scale features, as indicated in Fig. 3 (C1-C5). The sizes of these feature layers are 320 × 320, 160 × 160, 80 × 80, 40 × 40, and 20 × 20 pixels, respectively. As the network undergoes multiple downsampling operations, the C5 layer obtains a lower resolution but contains richer channel feature information. Therefore, EFFM is integrated into C5. In order to mitigate potential gradient issues and prevent any potential degradation of network performance during the deep feature extraction process, EFFM retains the residual structure from the original C2f module. This integration ensures the stability of gradients and maintains the overall network performance.

3.3 Global Attention Localization Module

The neck network of YOLOv8 combines the principles of the feature pyramid network (FPN) [30] and the path aggregation network (PAN) [31] for cross-scale feature fusion. As shown in Fig. 3, FPN strengthens the semantic features of the entire feature pyramid through the top-down pathway P_i and lateral connection L_i. However, it fails to transmit detailed location information from shallow layers to deep ones. To address this limitation, PAN complements FPN by introducing a bottom-up pathway N_i and lateral connection L_j, enabling the transmission of precise positional features from shallow layers to deep layers. This process improves the multi-scale localization capability.

In complex environments, vehicle detection poses challenges due to limited feature information for occluded and small target vehicles. Moreover, small target vehicles in the distance are highly influenced by environmental lighting and background noise. The shallow layer C2 contains the texture and edge features of the targets, providing more positional and detailed information. However, the original YOLOv8 sends the feature maps C3, C4, and C5 obtained from the backbone to the neck for feature

fusion, while excluding the shallowest feature map C2 with fewer semantic features. As the network deepens, operations like convolution and pooling can further deplete the already scarce information. Therefore, effective fusion of multi-scale features is crucial to address these issues and optimize the use of the available feature information.

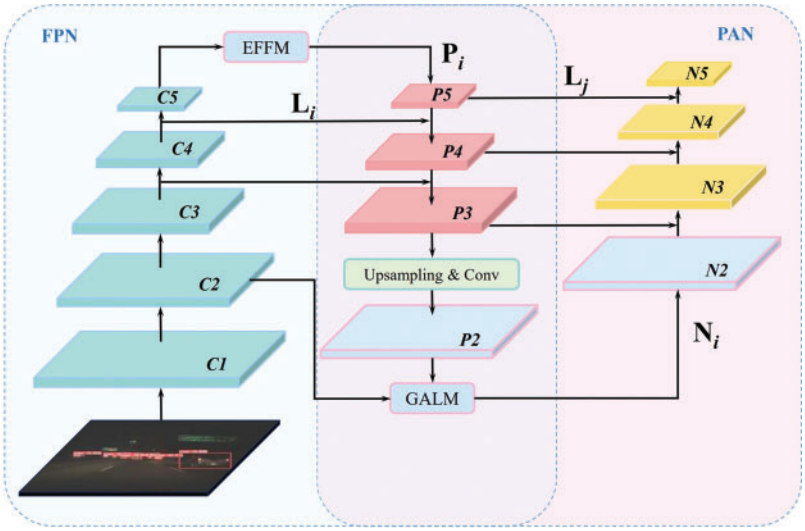

Figure 3: Feature fusion process of the FBPN

By fusing multi-scale features effectively, it becomes possible to incorporate both rich geometric details from shallow features and powerful semantic information from high-level features. This fusion process helps to retain important information and prevents loss during the feature fusion stage.

By performing an upsampling and convolution operation on P3, P2 was obtained that has a larger receptive field and stronger semantic information after being fused with Ci through FPN. As shown in Fig. 4, GALM draws inspiration from the global attention upsample (GAU) [32] module and uses global average pooling to provide global context that guides shallow features to select category localization information.

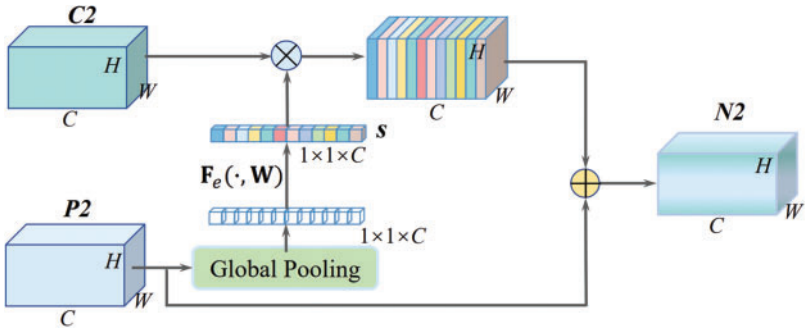

Figure 4: Structure of the GALM

In GALM, a global average pooling operation is performed on P2, generating channel-wise statistics. Formally, a statistic $Z \in R^C$ is generated by shrinking P2 through its spatial dimensions

$H \times W$, such that the c-th element of Z is calculated by [29]:

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} p2_c(i,j) \tag{2}$$

Further, the excitation operator is utilized to fully capture channel-wise dependencies. This operator allows for the adaptive assignment of different weights to the channels. The generated weight vector s is used to assign weights to the channels in C2 [29]:

$$s = F_e(z, W) = \sigma(W_1 \delta(W_{0z})) \tag{3}$$

where F_e represents the excitation mapping, W_0 is a fully connected layer with parameters W0, δ refers to the ReLU function, W_1 is a fully connected layer with parameters W1, σ is the sigmoid function.

In GALM, shortcut connections in residual structures are introduced to preserve semantic information of P2 without adding parameters and increasing computational complexity. Moreover, these connections enhance gradient values for backpropagation between layers, aiding in effective training of the network. After the feature fusion process, a new feature map N2 is generated with a size of $160 \times 160 \times 128$. This new feature map preserves more geometric and semantic information. Furthermore, by combining the original three detection heads, the newly generated four detection heads structure effectively improves the model's sensitivity to small vehicle features while mitigating the negative impact of severe scale changes on the object detector.

4 Experiments

4.1 Datasets and Implementation Details

Two mainstream autonomous driving datasets were used to evaluate the performance of the proposed model. The detailed information of these datasets is described as follows:

(1) BDD100K Dataset [33]: BDD100K is a dataset that is widely used for research and evaluation in autonomous driving. It consists of 10 classes with 100,000 images, covering various weather conditions, road traffic scenarios, and time periods of day-daylight, dusk, and night-time road scenes. For our experiment, we focused on three categories-Car, Bus, and Truck- and used a subset of 7,000 images with a total of 65,535 labels. Each image has a resolution of 1280 by 720 pixels, and we split the dataset into a training set (5,670 images), a validation set (630 images), and a test set (700 images) according to an 8:1:1 ratio.

(2) KITTI Dataset [34]: The KITTI dataset was jointly created by Toyota's American Technical Research Institute and the Karlsruhe Institute of Technology in Germany. It is currently one of the most representative datasets for autonomous vehicle detection, containing a total of 7481 actual road images captured by in-vehicle cameras in different environments such as urban, rural, and highways. For our experiment, we selected three categories-Car, Van, and Truck-out of eight categories and split the dataset into a training set (6058 images), a validation set (674 images), and a test set (749 images). This dataset is used to validate the generalization ability of the proposed method.

Windows operating system and PyTorch framework were used to implement the FBPN model proposed in this article. In the experiments, hyperparameters were configured as follows: a step decay learning rate schedule with an initial learning rate of 0.0001 was employed; momentum was set to 0.937 and weight decay to 0.0005. Additionally, to effectively showcase the structural benefits of the proposed model, we employed stochastic gradient descent (SGD) to optimize and update the model parameters, following the approach adopted in the baseline model. The input image size was set to 640×640 pixels for the BDD100K dataset and the KITTI dataset. Models are trained for 100 epochs

from scratch. To improve the model's robustness and address the issue of a limited number of instances in the "bus" and "truck" categories within the dataset, we employed the mosaic data augmentation technique. This technique, which has been utilized in the original YOLOv8, was applied consistently during the training process of all experiments. All experiments were conducted on hardware equipment consisting of an AMD Ryzen 9 7950X 16-core Processor (4.50 GHz) and an NVIDIA GeForce RTX 4090 graphics card.

4.2 Evaluating Indicators

This study adopts a set of widely used metrics [8] to evaluate the performance of the proposed FBPN. Precision (P) is defined as the percentage of correctly predicted positive cases in the sample, and recall (R) refers to the percentage of correctly predicted true positive cases in the sample. The calculation formulas for both P and R are as follows:

$$P = TP/(TP + FP) \tag{4}$$

$$R = TP/(TP + FN) \tag{5}$$

where TP, FP, and FN represent true positive, false positive, and false negative, respectively.

The area under the precision-recall curve, referred to as average precision (AP), which is a comprehensive metric for evaluating the detection performance, is used in this study to evaluate the detection performance for one single object category, denoted as:

$$AP = \int_0^1 P(R)\, dR \tag{6}$$

Consequently, mean average precision (mAP) is also employed as the evaluation metric, comprehensively reflecting both precision and recall performance, denoted as:

$$mAP = \frac{\sum_{i=1}^{n} AP_i}{n} \tag{7}$$

where n is the number of classes. mAP@0.5 represents the average AP of all categories when the IoU threshold is 0.5 and mAP@0.5:0.95 represents the average AP within a range. The IoU threshold ranges from 0.5 to 0.95 with a step size of 0.05.

In addition, the model's parameters and its giga floating point operations per second (GFLOPS) are calculated to evaluate the model's spatial and temporal complexity. Frames per second (FPS) is used to measure the detection speed of the models.

5 Results and Discussion

5.1 Detection Effects Analysis during Training

This study employed the BDD100K dataset to train YOLOv8 and the proposed FBPN. The curve of the total loss value during the training process is shown in Fig. 5. It is observed that FBPN exhibits a faster decrease in the total loss value compared to YOLOv8 for fewer than 10 training iterations. This indicates that FBPN has a superior ability to quickly learn and extract vehicle features in complex environments. Both algorithms exhibit insignificant decreases in the total loss value between 10 and 100 iterations, with the total loss curve stabilizing, signifying that both algorithms have effectively learned the vehicle features from the BDD100K training set.

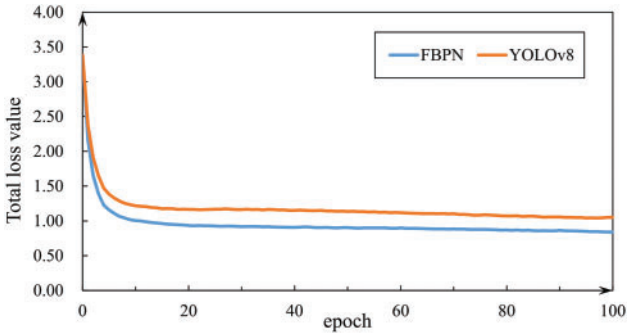

Figure 5: Total loss convergence curve

In addition, we compared the change curves of mAP@0.5 and mAP@0.5:0.95 during the training process. Figs. 6 and 7 show that the curve of FBPN is consistently higher than that of YOLOv8 in the middle and later stages of training. This suggests that FBPN possesses distinct advantages in the detection of vehicles in complex environments.

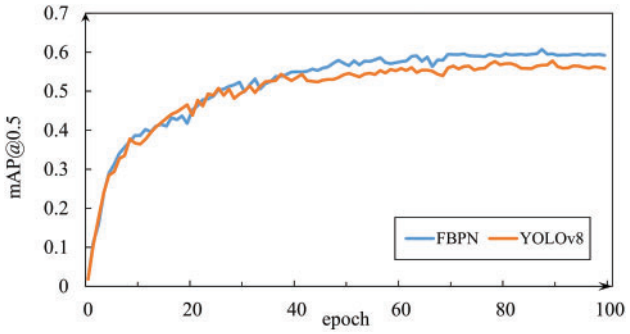

Figure 6: mAP@0.5

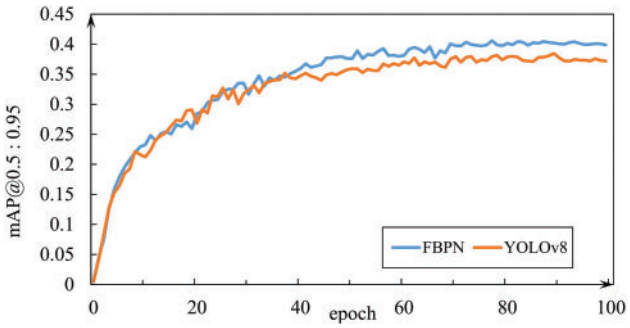

Figure 7: mAP@0.5:0.95

5.2 Ablation Studies of EFFM and GALM

To verify the effectiveness of the main components of the proposed method in vehicle detection tasks, a series of ablation experiments were conducted on the BDD100K dataset. Experimental results are shown in Table 1.

Table 1: Ablation study of EFFM and GALM

Method	Parameter (MB)	Model Size (MB)	GFLOPS	AP_{Car} (%)	AP_{Bus} (%)	AP_{Truck} (%)	mAP@0.5 (%)	FPS
YOLOv8 [11]	11.12	21.4	28.6	73.5	53.9	46	57.8	322.58
YOLOv8 + EFFM	11.12	21.4	**28.4**	72	54	47.4	57.8	312.50
YOLOv8 + GALM	11.33	21.7	32.7	73	54.9	46.6	58.2	318.60
YOLOv8 + GALM + 4-scale head	**10.63**	**20.6**	36.6	**77.1**	53.5	48.9	59.8	333.32
YOLOv8 + EFFM + GALM + 4-scale head (FBPN)	**10.63**	**20.6**	36.6	77	**56.3**	**49**	**60.8**	**344.83**

The results suggest that the inclusion of EFFM does not significantly impact the mAP value. However, the YOLOv8 benchmark model exhibits relatively low detection accuracy for the Bus and Truck categories, primarily due to the limited number of instances available in the dataset. In addition, vehicles in these categories possess a large aspect ratio, and the convolution and pooling operations used in the target detection algorithm can also cause loss of boundary features. Furthermore, larger vehicles are prone to be disconnected from their boundary features when disturbed by surrounding noise, resulting in misidentification as multiple targets. The EFFM module effectively addresses these challenges by expanding differences between target and background features while minimizing the increase in model parameters. As a result, it significantly enhances the detection accuracy for these two vehicle categories. This approach not only improves vehicle feature extraction and mitigates background interference but also reduces computational complexity. However, its integration into the deep network may unfortunately further reduce the already sparse information regarding small targets, resulting in decreased accuracy for the Car category.

The integration of GALM into YOLOv8 leads to a 0.4% increase in mAP. Furthermore, this improvement is further amplified to achieve a 59.8% mAP by integrating the four-scale detection head. Compared to the baseline algorithm, the GFLOPS has increased by 8G, while the parameter quantity and model size have decreased by 0.49 M and 0.8 G, respectively. Notably, the AP value for the Car category shows a more significant improvement, increasing by 3.6% compared to the baseline algorithm. This indicates that this integration effectively combines the semantic information contained in the deep network with the powerful fine-grained information from the shallow network. Furthermore, the additional detection head contains more fine-grained information of small target vehicles. Consequently, this integration significantly improves the utilization efficiency of feature information for occluded and small target vehicles.

The FBPN algorithm, with the addition of all three modules, achieves an mAP of 60.8%, demonstrating the highest detection performance. While reducing the number of parameters, the model detection rate also increases to 344.83 frames/s, and the AP values of the three types of vehicles are improved to certain degrees. The model size is 20.6 MB, making it more lightweight compared to the baseline method. This indicates that FBPN is advantageous in terms of computational resources. Overall, this algorithm utilizes local channel attention to suppress background and image noise on vehicle features while effectively fusing high-level and low-level features to maximize the utilization of rare feature information in complex environments. Moreover, it improves detection speed without compromising accuracy, enabling accurate and real-time vehicle detection in complex environments.

Furthermore, although the algorithm proposed in this paper demonstrates significant improvements over the baseline algorithm in terms of parameters, model size, accuracy, and speed, these advancements come at the cost of increased computational complexity. Traditional convolutions suffer from limitations in feature extraction and object localization due to their fixed scale and geometric structure [35]. These issues are addressed by introducing GALM. However, experimental results indicate that the introduction of GALM leads to increased computational load and an irregular memory access bottleneck, which may impose certain restrictions on widespread deployment on edge devices [36]. Therefore, in the future, our work will focus on selecting appropriate alternatives to traditional convolutions.

5.3 Comparison with Other Methods

To evaluate the accuracy and speed superiority of the proposed algorithm in complex vehicle detection environments, comparisons were conducted against several advanced vehicle image object detection algorithms using the BDD100K test set.

Table 2 presents the experimental results in detail, showing that the FBPN algorithm achieves the best overall performance with a map@0.5 of 60.8%. Furthermore, the AP of Car AP_{Car} reaches 77%, which has the highest average accuracy, indicating that the proposed method possesses significant advantages for detecting densely arranged vehicles and small target vehicles in complex backgrounds. It is worth noting that the FPS value of FBPN reached 344.83 frames/s, which is the highest detection speed among these advanced algorithms.

Table 2: Performance comparison of proposed method and other detectors on BDD100K dataset

Method	AP_{Car} (%)	AP_{Bus} (%)	AP_{Truck} (%)	mAP@0.5 (%)	FPS
YOLOv4 [37]	61.35	56.92	53.71	57.32	41.26
DR-YOLOv4 [37]	62.23	58.07	**54.42**	58.24	43.50
YOLOv5s [38]	71.40	55.50	48.90	58.60	66.67
CAM-YOLO [38]	76.20	**58.70**	47.00	60.60	58.82
YOLOv7-tiny [39]	70.00	42.10	41.10	51.10	312.50
RT-DETR [40]	**77.00**	52.40	48.40	59.27	243.90
Proposed	**77.00**	56.30	49.00	**60.80**	**344.83**

We exhibit the detection findings in complex road scenarios, as displayed in Fig. 8, to qualitatively evaluate the performance of the proposed algorithm.

FBPN reduces the amount of false and missed alerts for vehicle detection in complex backgrounds, as displayed in Fig. 8. In the first image, the vehicle within the green box is unlabeled, which could explain the failure of YOLOv7-tiny to detect it. However, both YOLOv8 and FBPN models successfully identified the vehicle. In the yellow circled area, YOLOv8 and YOLOv7-tiny encountered challenges in accurately distinguishing between the background and target objects, leading to the misidentification of surrounding objects as cars. Moreover, in the third image, small target vehicles face challenges due to illumination complexities and serious occlusions between vehicles, especially in low-light conditions where the image quality is poor. In such scenarios, the vehicle contours become almost indistinguishable. Particularly in the fourth image, FBPN remains highly effective, but YOLOv8 and

YOLOv7-tiny have shown poor performance with varying degrees of missed detections and false detections.

Figure 8: The qualitative detection outcomes of various methods on the BDD100K dataset

One possible reason is that the proposed algorithm in this study expands the feature difference between the vehicle and the background by using a local channel attention mechanism. Meanwhile, GALM was utilized to enhance the model's localization ability and improve its sensitivity to small vehicles. The combination of these two enables the model to exhibit strong anti-interference capabilities in complex environments. The proposed algorithm demonstrates effectiveness in various complex scenarios, even in cases of image blurriness, significantly reducing the occurrence of missed detections and false detections.

5.4 Performance on the KITTI Dataset

To further validate the effectiveness of FBPN in other complex road scenarios, a series of experiments were performed on the KITTI dataset comparing FBPN with three series of algorithms, namely Faster-RCNN, SSD, and YOLO. As shown in Table 3, the Faster R-CNN and SSD algorithms show relatively low mAP and FPS, making them inadequate for practical detection scenarios. Our proposed model demonstrates suboptimal performance for specific vehicle categories, namely Bus and Truck. These vehicles exhibit a relatively large aspect ratio and have narrow feature information in certain parts, which makes the feature information in these areas more susceptible to fragmentation. Additionally, the repeated application of convolution and pooling operations contributes to the loss of fine-grained feature information in these narrow areas. Despite a slight decrease in overall accuracy (0.41%) compared to the top-performing DR-YOLOv4, our algorithm achieves a detection speed that is 7.5 times faster than DR-YOLOv4. Remarkably, the mAP for the Car category reaches an impressive 98.20%, surpassing the performance of existing state-of-the-art detectors. These results

indicate that the proposed FBPN achieves a favorable trade-off between accuracy and speed, making it more suitable for subsequent embedded development applications. Consequently, the comprehensive performance of this algorithm on the KITTI dataset has been further verified.

Table 3: Performance comparison of the proposed method and other detectors on the KITTI dataset

Method	AP_{Car} (%)	AP_{Bus} (%)	AP_{Truck} (%)	mAP@0.5 (%)	FPS
Faster R-CNN [37]	78.25	72.93	80.09	77.09	12.47
SSD300 [37]	81.59	74.82	87.36	81.26	56.71
SSD512 [37]	84.81	70.72	82.98	79.50	26.24
YOLOv4 [37]	95.68	97.72	**98.83**	97.41	45.72
DR-YOLOv4 [37]	97.48	**98.51**	98.65	**98.21**	48.05
YOLOv5s [38]	97.90	97.50	97.50	97.60	90.91
CAM-YOLO [38]	**98.20**	98.30	97.90	98.13	76.92
YOLOv7-tiny [39]	92.20	93.90	89.40	93.20	278.45
YOLOv8 [11]	97.20	96.80	98.30	97.40	303.03
RT-DETR [40]	97.60	95.90	96.80	96.80	256.41
Proposed	**98.20**	97.20	98.00	97.80	**357.14**

6 Conclusions

This paper proposes a novel vehicle detection algorithm called FBPN, from the perspective of drivers. A scheme was designed that enhances the extraction and fusion of vehicle feature information from the channel dimension. Firstly, the EFFM module is used to obtain long-range interdependence between channels, and the generated channel attention weights are used to enhance the feature expression of important channels while avoiding the consumption of extensive computational resources. Secondly, the GALM module incorporates shallow high-resolution features into the pyramid information transfer to effectively improve the detector's perception ability of small target detail position information. Finally, the four-layer feature pyramid structure retains the geometry and semantic information favorable to small targets to the greatest extent, effectively coping with the adverse effects brought by target scale changes. Experimental results on publicly available BDD100K and KITTI datasets show that the proposed algorithm achieved average accuracies of 60.80% and 97.80%, respectively, with inference speeds reaching 344.83 frames/s and 357.14 frames/s. Compared to other algorithms, this proposed algorithm has relatively comprehensive accuracy and inference speed, providing efficient detection performance in complex environments. In our future work, our goal is to further reduce the computational complexity of the model while maintaining the same accuracy and speed. Additionally, we aim to deploy the algorithm model for real-time vehicle detection on edge devices.

Acknowledgement: The authors would like to thank the editors and the anonymous reviewers for their valuable comments that greatly improved our manuscript.

Funding Statement: This research was funded by Ministry of Science and Technology of the People's Republic of China, Grant Numbers 2022YFC3800502; Chongqing Science and Technology Commission, Grant Number cstc2020jscx-dxwtBX0019, CSTB2022TIAD-KPX0118, cstc2020jscx-cylhX0005 and cstc2021jscx-gksbX0058.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: D. Tianmin, Z. Xiyue; data collection: Z. Xiyue; analysis and interpretation of results: D. Tianmin, Z. Xiyue. C. Xinxin; draft manuscript preparation: Z. Xiyue. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data available on request from the authors. The data that support the findings of this study are available from the corresponding author, D. Tianmin, upon reasonable request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] L. Kang, H. Shen, Y. Li and S. Xu, "A data-driven control-policy-based driving safety analysis system for autonomous vehicles," *IEEE Internet of Things Journal*, vol. 10, no. 16, pp. 14058–14070, 2023.
[2] B. Mahaur and K. Mishra, "Small-object detection based on YOLOv5 in autonomous driving systems," *Pattern Recognition Letters*, vol. 168, pp. 115–122, 2023.
[3] Z. Li, Y. Du, M. Zhu, S. Zhou and L. Zhang, "A survey of 3D object detection algorithms for intelligent vehicles development," *Artificial Life and Robotics*, vol. 27, pp. 115–122, 2022.
[4] C. Ma, J. Song, Y. Xu, H. Fan, X. Wu *et al.,* "Vehicle-based machine vision approaches in intelligent connected system," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2023.
[5] J. D. Trivedi, S. D. Mandalapu and D. H. Dave, "Vision-based real-time vehicle detection and vehicle speed measurement using morphology and binary logical operation," *Journal of Industrial Information Integration*, vol. 27, pp. 100280, 2022.
[6] L. Wang, H. Zhong, W. Ma, M. Abdel-Aty and J. Park, "How many crashes can connected vehicle and automated vehicle technologies prevent: A meta-analysis," *Accident Analysis & Prevention*, vol. 136, pp. 15299, 2020.
[7] K. Muhammad, A. Ullah, J. Lloret, J. D. Ser and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4316–4336, 2021.
[8] J. Karangwa, J. Liu and Z. Zeng, "Vehicle detection for autonomous driving: A review of algorithms and datasets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 11568–11594, 2023.
[9] Y. Song, S. Hong, C. Hu, P. He, L. Tao *et al.,* "MEB-YOLO: An efficient vehicle detection method in complex traffic road scenes," *Computers, Materials & Continua*, vol. 75, no. 3, pp. 5761–5784, 2023.
[10] L. Kang, Z. Lu, L. Meng and Z. Gao, "YOLO-FA: Type-1 fuzzy attention based YOLO detector for vehicle detection," *Expert Systems with Applications*, vol. 237, pp. 121209, 2024.
[11] Ultralytics. [Online]. Available: https://github.com/ultralytics/ultralytics/ (accessed on 01/02/2023)
[12] A. Boukerche and Z. Hou, "Object detection using deep learning methods in traffic scenarios," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–35, 2021.
[13] J. Azimjonov and A. Özmen, "A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways," *Advanced Engineering Informatics*, vol. 50, pp. 101393, 2021.
[14] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. of CVPR*, Columbus, OH, USA, pp. 580–587, 2014.

[15] R. Girshick, "Fast R-CNN," in *Proc. of ICCV*, Santiago, Chile, pp. 1440–1448, 2015.
[16] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2022.
[17] R. Ghosh, "On-road vehicle detection in varying weather conditions using faster R-CNN with several region proposal networks," *Multimedia Tools and Applications*, vol. 80, no. 17, pp. 25985–25999, 2021.
[18] P. Gao, T. Tian, T. Zhao, L. Li and J. Tian, "Double FCOS: A two-stage model utilizing FCOS for vehicle detection in various remote sensing scenes," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 4730–4743, 2022.
[19] J. Q. Luo, H. S. Fang, F. M. Shao, Y. Zhong and X. Hua, "Multi-scale traffic vehicle detection based on faster R–CNN with NAS optimization and feature enrichment," *Defence Technology*, vol. 17, no. 4, pp. 1542–1554, 2021.
[20] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. of CVPR*, Las Vegas, NV, USA, pp. 779–788, 2016.
[21] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed *et al.,* "Ssd: Single shot multibox detector," in *Proc. of ECCV*, Amsterdam, The Netherlands, pp. 21–37, 2016.
[22] Z. Chen, H. Guo, J. Yang, H. Jiao, T. Gao *et al.,* "Fast vehicle detection algorithm in traffic scene based on improved SSD," *Measurement*, vol. 201, pp. 111655, 2022.
[23] T. Deng, X. Liu and L. Wang, "Occluded vehicle detection via multi-scale hybrid attention mechanism in the road scene," *Electronics*, vol. 11, no. 17, pp. 2709, 2022.
[24] M. Bie, Y. Liu, G. Li, J. Hong and J. Li, "Real-time vehicle detection algorithm based on a lightweight you-only-look-once (YOLOv5n-L) approach," *Expert Systems with Applications*, vol. 213, pp. 119108, 2023.
[25] H. Chen, H. Jin and S. Lv, "YOLO-DSD: A YOLO-based detector optimized for better balance between accuracy, deployability and inference time in optical remote sensing object detection," *Applied Sciences*, vol. 12, no. 15, pp. 7622, 2022.
[26] J. Ye, Z. Yuan, C. Qian and X. Li, "CAA-YOLO: Combined-attention-augmented YOLO for infrared ocean ships detection," *Sensors*, vol. 22, no. 10, pp. 3782, 2022.
[27] J. Zheng, T. Wang, Z. Zhang and H. Wang, "Object detection in remote sensing images by combining feature enhancement and hybrid attention," *Applied Sciences*, vol. 12, no. 12, pp. 6237, 2022.
[28] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo *et al.,*, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. of CVPR*, USA, pp. 11534–11542, 2020.
[29] J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks," in *Proc. of CVPR*, Salt Lake City, UT, USA, pp. 7132–7141, 2018.
[30] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan *et al.,* "Feature pyramid networks for object detection," in *Proc. CVPR*, Honolulu, HI, USA, pp. 2117–2125, 2017.
[31] S. Liu, L. Qi, H. Qin, J. Shi and J. Jia, "Path aggregation network for instance segmentation," in *Proc. of CVPR*, Salt Lake City, UT, USA, pp. 8759–8768, 2018.
[32] H. Li, P. Xiong, J. An and L. Wang, "Pyramid attention network for semantic segmentation," arXiv Preprint, arXiv:1805.10180, 2022.
[33] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen *et al.,* "BDD100K: A diverse driving dataset for heterogeneous multitask learning," in *Proc. of CVPR*, Seattle, WA, USA, pp. 2636–2645, 2020.
[34] A. Geiger, P. Lenz and R. Urtasun, "Are we ready for autonomous driving? The kitti vision benchmark suite," in *Proc. of CVPR*, Providence, RI, USA, pp. 3354–3361, 2012.
[35] W. Li, B. Li, C. Yuan, Y. Li and F. Wang, "Anisotropic convolution for image classification," *IEEE Transactions on Image Processing*, vol. 29, pp. 5584–5595, 2020.
[36] J. Guan, R. Lai, Y. Lu, Y. Li and L. Gu, "Memory-efficient deformable convolution based joint denoising and demosaicing for UHD images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 11, pp. 7346–7358, 2022.

[37] T. Deng, G. Mao, Z. Zhou and Z. Duan, "Road vehicle detection and recognition algorithm based on densely connected convolutional neural network," *Journal of Computer Applications*, vol. 42, no. 3, pp. 883, 2022.
[38] T. Deng, X. Liu, L. Wang and C. Wang, "Vehicle detection algorithm combined with cascading attention mechanism," *Computer Engineering and Applications*, vol. 59, no. 21, pp. 141–150, 2023.
[39] C. Y. Wang, A. Bochkovskiy and H. Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. of CVPR*, Vancouver, BC, Canada, pp. 7464–7475, 2023.
[40] W. Lv, S. Xu, Y. Zhao, G. Wang, J. Wei *et al.,* "DETRs beat YOLOs on real-time object detection," arXiv Preprint, arXiv:2304.08069, 2023.