BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 73(5), 2025, Article number: e155037
DOI: 10.24425/bpasts.2025.155037

Combining improved generative adversarial networks for end-to-end traffic object detection under complex illumination conditions

Yang LIU[1], **Zhe GONG**[2], **Yuyang HE**[1]*, and **Weiqin LI**[3]

[1] Jiangsu Vocational College Of Information Technology, School of Automobile and Intelligent Traffic, Wuxi 214153, China
[2] China National Offshore Oil Corporation, CNOOC Research Institute Company Limited, Beijing 100027, China
[3] Beijing Benz Automotive Co., Ltd., Beijing 100176, China

Abstract. The images captured by vehicle-mounted cameras in low-illumination environments have the problem of severe loss of detailed information. At the same time, the detection and recognition performance of traffic object detection algorithms is also influenced by factors such as object texture, movement speed, shooting angle, and occlusion. Under low-illumination conditions, the background of images is integrated with traffic objects, so the current object detection algorithms have relatively poor performance in detecting traffic objects under low illumination. In order to achieve low-illumination image enhancement without significantly reducing the reasoning speed of object detection algorithms and meanwhile improve the detection accuracy of object detection algorithms under low-illumination conditions, a multi-object detection model based on image enhancement, namely low-illumination enhancement and deep fusion-you only look once (LEDF-YOLO), is proposed. Firstly, based on the generative adversarial network (GAN) model, the direct-to-deep-generative adversarial network (DD-GAN) model is proposed to improve the effect of enhancing low-illumination images. Then, the fusion and parallel-cross stage partial bottleneck with two convolutions (FP-C2f) module and the transformer-spatial pyramid pooling fast (T-SPPF) module were designed to enhance and fuse multi-scale features. Finally, the network model of you only look once version 8n (YOLOv8n) was improved by introducing cross-hierarchical connections, making object localization more accurate. Experimental results on UA-DETRAC and self-made datasets showed that compared to the YOLOv8n algorithm, the LEDF-YOLO object detection method improved detection accuracy while maintaining the high real-time performance of the you only look once version 8n (YOLOv8n) algorithm.

Keywords: low-illumination; object detection; object recognition; image enhancement; autonomous driving.

1. INTRODUCTION

Autonomous driving technology can improve the safety of drivers and pedestrians and reduce casualties and property losses [1]. With the rapid development of artificial intelligence (AI) technology, object detection methods based on deep learning (DL) are widely applied in the field of autonomous driving and provide good safety guarantees for autonomous vehicles [2]. One of the essential ways for autonomous driving systems to perceive the environment is to obtain environmental images around vehicles through onboard cameras to detect traffic objects [3–5]. However, when driving in low-illumination conditions, the performance of autonomous driving systems is currently poor due to the weak exposure of onboard optical sensors, interference light sources in the environment, and uneven illumination. If autonomous driving technology is to be widely popularized in all application areas, it is necessary to make corresponding improvements to object detection technology under low-illumination conditions, that is, to meet the performance requirements of traffic object detection and real-time (the detection speed reaches 30 frames per second (FPS) [6]) image processing under low-illumination conditions [7, 8]. Therefore, autonomous driving technology under low-illumination conditions based on object detection has significant research significance for achieving high-level autonomous driving.

In response to the above shortcomings, this study proposes a traffic object detection method with high detection accuracy and the ability to output real-time detection results in various illumination images, which can be used under low-illumination conditions. Firstly, this algorithm combines the data advantages of traffic images and the strong feature extraction ability of deep learning networks to propose the DD-GAN algorithm to enhance low-illumination images. Secondly, the FP-C2f module and T-SPPF module were proposed to enhance and fuse features to improve the feature extraction ability for small objects, and the structure of the YOLOv8n algorithm has been redesigned to extract features of small objects effectively in image or video frames. At the same time, in response to the problem of a single scene and small differences in object size in the current mainstream low-illumination traffic object detection dataset, this study proposes a traffic object detection dataset that includes images of various illumination scenes and is named toward low-light or night-Datest (TLLN-Datest). Compared with the YOLOv8n algorithm, the LEDF-YOLO (see Fig. 1) algo-

*e-mail: 2024101365@jsit.edu.cn

Manuscript submitted 2025-01-21, revised 2025-05-24, initially accepted for publication 2025-07-02, published in August 2025.

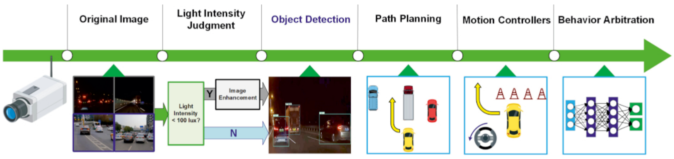


Fig. 1. Overview of the role of the proposed traffic object detection method in autonomous driving systems. Based on relevant research results, when the natural illumination is below 100 lux, the driver's perception of the environment significantly decreases [10, 11]

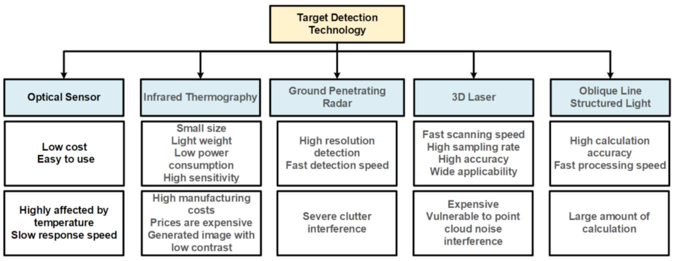
rithm significantly improves the detection accuracy of traffic objects on both the public dataset named UA-DETRAC [9] and the TLLN-Datest was proposed in this study while meeting the speed requirements for object detection. It can effectively detect large and small objects in real time, with good multi-scale detection performance. The main contributions of this study are as follows:

- In order to solve the problem of small traffic objects in low-illumination conditions being challenging to detect, the improved algorithm proposed the DD-GAN image enhancement method.
- In view of the significant difference in the size of traffic objects, we proposed a network modules that integrate the attention mechanism, named FP-C2f, to extract and fuse the features of different scales effectively to improve detection accuracy.
- In response to the challenge of context information loss within network structures, which hampers the detection of small objects, we have proposed the T-SPPF module, which ensures that the object position information is not lost while keeping the original receptive field unchanged. This network facilitates the integration of contextual feature information, strengthening the contextual associations of small objects and bolstering the detection capabilities of the object detection algorithm.
- A new dataset, TLLN-Datest, is made for the various categories and large differences in the size of traffic objects, which can meet the common traffic object detection tasks under low-illumination conditions. To our knowledge, the TLLN dataset is currently one of the largest datasets used for object detection, consisting of both low-illuminance images and normal images.
- The LEDF-YOLO algorithm has conducted a large number of experiments on the TLLN-Datest and public dataset, which verifies the superiority of the LEDF-YOLO algorithm under complex illumination conditions.

2. RELATED WORK

2.1. Innovations in object detection

With the rapid development of DL, object detection methods based on DL dominate in object detection [12, 13]. Currently, most mainstream object detection algorithms for detecting traffic objects are based on normal illumination conditions and trained using a universal dataset with more normal conditions. However, if object detection algorithms that can achieve good results in limited scenes are directly applied to images or videos captured under low-illumination conditions, they usually yield poor detection effects [14]. This is because images or videos captured in low-illumination environments often have characteristics such as low signal-to-noise ratio and low contrast, which can result in sparse semantic features of traffic objects in the image. Therefore, the applicability of general object detection algorithms to low-illumination images is poor, which leads to poor overall detection performance of object detection algorithms in traffic scenes. For RGB images, Loh *et al*. [15] proposed a dedicated dataset, named ExDark, for low-illumination object detection, which analyzed and improved the detection performance of manual feature extraction methods and hoped to promote research related to low-illumination object detection. Sasagawa *et al*. [16] designed the YOLO-in-the-Dark network model, which combines pre-trained models from different domains using adhesive layers and generation models, namely the low-illumination image enhancement model [17] and the object detection model YOLO [18], thereby improving the detection performance of low-illumination object detection algorithm. Tao *et al*. [18] used low-resolution and low-illumination image enhancement networks to reduce the computational complexity of object detection algorithms. The generated images were then transmitted to the EfficientDet detection network model through a super-resolution network to obtain detection results, which improved the reliability of object detection algorithms. Chen *et al*. [19] proposed a low-illumination image enhancement network model, which is more suitable for object detection from the perspective of feature retrieval, and combined it with single shot multibox detector (SSD) [20] detection network to improve object detection accuracy. At present, low illumination object detection algorithms based on domain adaptation require the use of paired datasets of light and dark to train GAN [21], and then the detection results are output by a universal object detection algorithm. The traffic environment perception technology based on the optical sensor has the advantage of rich semantic information of the generated image and is low-cost (see Fig. 2). We used it in image recognition and object detection systems. However, current object detection algorithms under low-illumination conditions have not effectively overcome the defect of unclear detail features in low-illumination images and have not fully extracted and utilized limited features to output high-precision detection results.



Optical Sensor	Infrared Thermography	Ground Penetrating Radar	3D Laser	Oblique Line Structured Light
Low cost Easy to use	Small size Light weight Low power consumption High sensitivity	High resolution detection Fast detection speed	Fast scanning speed High sampling rate High accuracy Wide applicability	High calculation accuracy Fast processing speed
Highly affected by temperature Slow response speed	High manufacturing costs Prices are expensive Generated image with low contrast	Severe clutter interference	Expensive Vulnerable to point cloud noise interference	Large amount of calculation

Fig. 2. The composition of object detection technology

2.2. The YOLOv8 object detection algorithm

As the latest YOLO algorithm, YOLOv8 can be applied in object detection, image classification, and instance segmentation tasks. This study selected YOLOv8n (the network structure of the

YOLOv8n module can be seen in Fig. 3) with a small parameter quantity and high detection accuracy based on the consideration of running algorithms on ordinary devices (see Table 1 and Table 2). The network model of the YOLOv8n algorithm mainly consists of four parts: Input, Backbone, Neck, and Head. In order to enable the YOLOv8n algorithm to extract the semantic features while ensuring it is lightweight, YOLOv8n has designed a new C2f structure. In this study, an improved algorithm based on YOLOv8, LEDF-YOLO, is proposed to solve the problems of multiple types, small size, and being easily occluded by other objects in the traffic scene. Based on the current advanced one-stage object detector YOLOv8, this algorithm has made several improvements in network structure and modules to adapt to complex and challenging traffic object detection tasks.

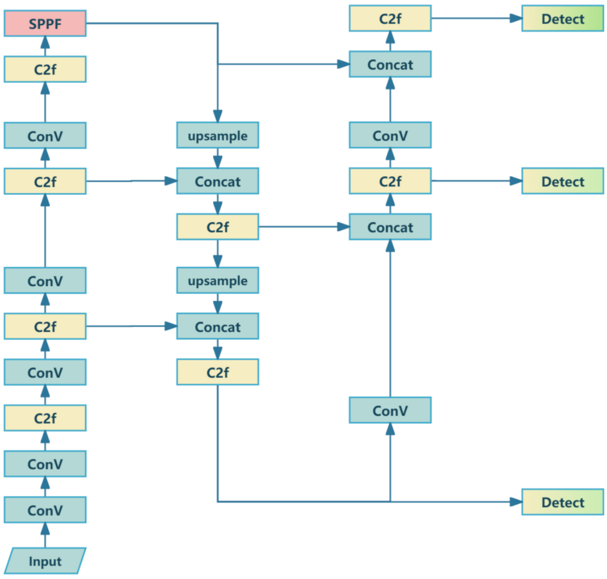


Fig. 3. The network structure of the YOLOv8n module

2.3. The standard components and functions of YOLOv8 and GAN

The Backbone part of the YOLOv8 network model is responsible for feature extraction, using a series of convolutional layers, as well as residual connections and bottleneck structures to reduce network size and improve performance. The YOLOv8 network model uses the C2f module as the basic building block, which has fewer parameters and better feature extraction capabilities compared to the C3 module of the YOLOv5 network model. Specifically, the C2f module reduces redundant parameters and improves computational efficiency through more efficient struc-

Table 1

Comparison between the YOLOv8 series object detection methods on a single RTX 3070Ti using the MS-COCO dataset [22]. The MS-COCO dataset is currently one of the most widely used small object detection datasets. It has over 330K images and 2.5M annotations, including 22 categories, such as humans, animals, and transportation. The characteristics of the MS-COCO dataset are a high proportion of small objects, and the image content in the dataset is complex and diverse. In recent years, the MS-COCO dataset has been used to organize a series of small object detection competitions. The optimal value for each of these evaluation criteria has been bolded in black, "↑" indicates that a larger value of the evaluation criterion is better, while "↓" indicates that a smaller value of the evaluation criterion is better

Algorithm	Parameters(M)↓	FLOPs(B)↓	Speed of reasoning(CPU/GPU)	mAP@0.5:0.95(%)↑	The area of application
YOLOv8n	**3.2**	**8.7**	**Fast**	37.3	Mobile/Edge devices
YOLOv8s	11.1	28.6	Medium	44.9	Balance speed and accuracy
YOLOv8m	25.9	78.7	Slower	50.2	General scenarios
YOLOv8l	43.7	165.4	Slow	52.9	High detection precision requirements
YOLOv8x	68.2	257.8	Slowest	**53.9**	Adequate computing resources

Table 2

Comparison between the mainstream object detection algorithms on a single RTX 3070Ti using the Microsoft COCO dataset [22] and the PASCAL VOC dataset [23]. It can be seen that compared with some mainstream algorithms, the YOLOv8n algorithm has basically maintained a leading position in various evaluation indexes, so this study improves on the basis of the YOLOv8n algorithm. "↑" indicates that a larger value of the evaluation criterion is better, while "↓" indicates that a smaller value of the evaluation criterion is better

Algorithm	FLOPs(G)↓	mAP COCO(%)↑	mAP VOC(%)↑	Size	Parameters(M)↓	Model Size(MB)↓
YOLOv8n	8.7	62.3	81.7	640	3.2	6.3
SSD 300 [20]	61.2	62.8	84.0	300	24.1	93.1
YOLOv6s [24]	27.4	59.6	76.8	640	10.6	81.3
YOLOv8s	28.6	66.1	84.2	640	11.1	22.5
YOLOv7-tiny [25]	13.1	65.2	83.8	640	6.0	12.2
YOLOv4-tiny [26]	6.8	56.7	63.3	416	5.9	22.5
YOLOX-s [27]	26.8	63.4	84.1	640	8.9	34.3
YOLOv5s	15.8	60.2	82.0	640	7.0	14.4

tural design. The Neck part of the YOLOv8 network model is responsible for multi-scale feature fusion, enhancing feature representation capability by fusing feature maps from different stages of Backbone. Specifically, the Neck part of the YOLOv8 network model includes the following components:

- The spatial pyramid pooling fast (SPPF) module: The SPPF module is used for pooling operations at different scales, concatenating feature maps of different scales together to improve the detection ability of objects of different sizes.
- The path aggregation network (PAN) module: The YOLOv8 network model consists of two PAN modules used for path aggregation of features at different levels, enhancing the expressive power of feature maps through bottom-up and top-down paths.

The Head part of the YOLOv8 network model is responsible for the final object detection and classification tasks, including a detection head and a classification head:

- The detection head module: It includes a series of convolutional layers used to generate detection results. These layers are responsible for predicting the bounding box regression values of each anchor box and the confidence level of the existence of the object.
- The classification head module: Global average pooling is used to classify each feature map, and the probability distribution of each category is output by reducing the dimensionality of the feature map. The design of the classification header enables YOLOv8 to effectively handle multi-class classification tasks.

The CBS module (see Fig. 4a) in the GAN network model includes a convolutional layer (Conv), a batch normalization layer (BN), and a SiLU activation function, while the CBL module (see Fig. 4b) includes a convolutional layer (Conv), a batch normalization layer (BN), and LeakyReLU activation function. The detection head module and the classification head module are mainly used for feature extraction. Meanwhile, introducing the nonlinear factor of the activation function enables the model to learn more complex mapping relationships.

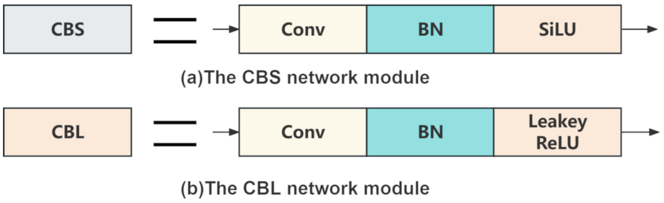


Fig. 4. The network structure of the CBS module and the CBL module

3. THE METHODS

In this study, the traffic object detection algorithm is trained with the relevant traffic detection dataset. The marked video frames output by the LEDF-YOLO algorithm can be regarded as the input state of the deep reinforcement learning (DRL) method. This section mainly introduces the background and component modules of the LEDF-YOLO method.

3.1. The DD-GAN module

The feature extraction module of the U-shaped network of the EnlightenGAN network model can get high-level features with rich detail information and shallow features with rich semantic information. This module design aims to combine the advantages of high-level features and shallow features to reduce information loss during feature transmission. However, the EnlightenGAN [28] method did not effectively integrate multi-scale features. In order to address this problem, we proposed the DD-GAN method based on the EnlightenGAN method, which can be used in the preprocessing step of the YOLO object detection algorithm (see the image enhancement module of Fig. 1). In order to maintain the detailed information of the original image to the greatest extent possible, the convolution kernel size of all convolutional layers is set to 3×3, and the stride is set to 1. Meanwhile, the residual modules are introduced to preserve as much of the colour and texture information of the original image as possible. The attenuation of light during propagation and the reflection and absorption of the object can cause uneven illumination distribution on different image pixels. Therefore, spatial attention is used to fully utilize the different position information of illumination distribution in the feature map to fit the distribution representation of low-illumination images to the distribution representation of normal illumination images. The convolutional block attention module (CBAM) [29] utilizes different information through two methods: average pooling and maximum pooling. Maximum pooling encodes the most significant part of the illumination distribution in low-illumination images, while average pooling encodes the global semantics information of low-illumination images. Therefore, using the CBAM module simultaneously can allow generated images to fit the illumination distribution of the normal illumination image better. The network model of the DD-GAN method is shown in Fig. 5.

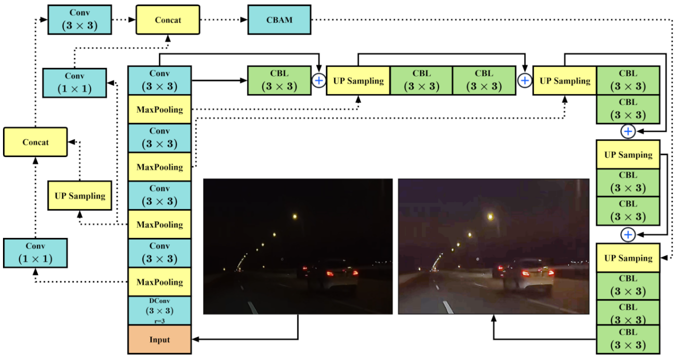


Fig. 5. The main components and steps of the DD-GAN module

In DD-GAN, training a large number of images enables the network to acquire the ability to process low-illumination images. Firstly, a set of images is selected, including a low-illumination image and an unpaired normal illumination image. During training, the low-illumination image is input into the generator, and the enhanced image is output. The difference between the output image and the normal illumination image is calculated based on the loss function, and the parameters of the generator are adjusted. At the same time, the enhanced image and the normal illumination image are inputted to the global and local discriminators for discrimination. The discriminator extracts image features, calculates the difference between the enhanced image and the normal illumination image through a loss function, and feeds it back to the generator to improve continuously. After repeating this process many times, the generator and discriminator reach a balance through training and, finally, obtaining the generator with the best parameters. The DD-GAN method can convert the low-illumination traffic image into a normal image, providing more significant semantic feature information for subsequent traffic object detection tasks.

Image enhancement typically uses peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) as quantitative indicators. PSNR measures the overall enhancement effect by evaluating the pixel difference between the generated enhanced image and the normal illumination image in decibels (dB). The formula is as follows:

$$PSNR = 10 \times \log_{10}\left(\frac{\max_I^2}{\mathrm{MSE}}\right), \tag{1}$$

where $\max_I^2$ represents the maximum value of the pixel.

SSIM balances the structural differences between images to represent the degree of detail similarity in the image. The formula is shown in 2.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2))}{(\mu_x^2\mu_y^2 + c_1)(\sigma_x^2\sigma_y^2 + c_1)}, \tag{2}$$

where μ_x is the average of x, μ_y is the average of y, σ_x^2 is the variance of x, σ_y^2 is the variance of y, and σ_{xy} is the covariance of x and y. $c_1 = (k_1L)^2$,$c_2 = (k_2L)^2$ is a constant used to maintain stability. L is the dynamic range of pixel values. $k_1 = 0.01$, $k_2 = 0.03$.

In order to verify the enhancement effect of the DD-GAN algorithm proposed in this study, the training set was selected from 1930 unpaired image datasets provided by the EnlightenGAN algorithm, including 934 low-illumination images and 1016 unpaired normal low-illumination images. The images in the test set are sourced from the public low-illumination dataset DICM [30], MEF [31], and ExDark [15], where the ExDark dataset only contains 157 randomly selected images. To verify the performance of the proposed method on paired image datasets, 148 low-illumination/normal-illumination image pairs were randomly selected from the dataset used to obtain training data to validate the algorithm in this study. All image sizes have been adjusted to 600×400 pixels. From Fig. 6 and Table 3, we can see that the image enhancement effect of the DD-GAN algorithm has significantly improved compared to the EnlightenGan algorithm.

3.2. The T-SPPF module

In autonomous driving scenarios, there are often incomplete traffic objects caused by factors such as occlusion between traffic objects in the image, resulting in low detection accuracy of object detection algorithms. The SPPF module can obtain semantic features of local receptive fields and approximate non-local receptive fields, fuse semantic features of different scale receptive fields, and enrich the expression ability of feature maps. However, the feature map will lose the position information of the object after the maximum pooling operation in the SPPF module. Therefore, the dilated convolution [32] operation is used to replace the maximum pooling operation in the T-SPPF (see Fig. 7) module, ensuring that the object position information is not lost while keeping the original receptive field unchanged. Dilated convolution introduces a parameter to define the dilated rate of the convolution, which can obtain local and nonlocal information without increasing the number of convolution kernels to obtain more object position information.

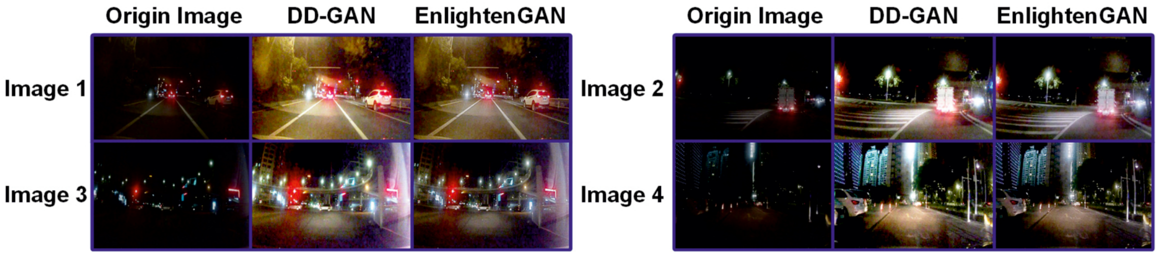


Fig. 6. Visual comparison with relevant image enhancement methods

Table 3

The objective evaluation indicators for several mainstream image enhancement methods on the self-made road lane dataset. The optimal value for each of these evaluation criteria has been bolded in black, "↑" indicates that a larger value of the evaluation criterion is better, while "↓" indicates that a smaller value of the evaluation criterion is better

Indicator	Zero-DCE [33]	StableLLVE [34]	KinD [35]	EnlightenGan [28]	DDGAN
PSNR ↑	11.26	14.19	15.76	14.37	**16.19**
SSIM ↑	0.71	0.82	0.90	0.85	**0.92**

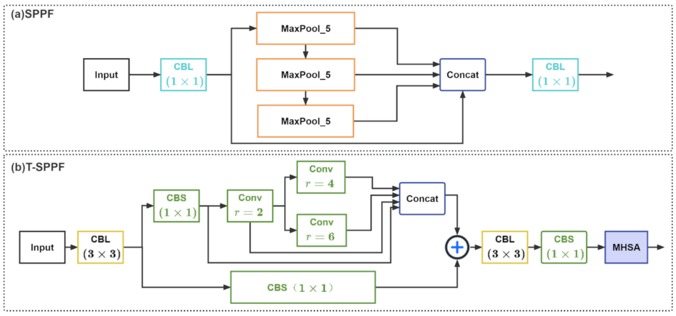


Fig. 7. (a) The network structure of the SPPF module. (b) The network structure of the T-SPPF module. The T-SPPF module accelerates network convergence by introducing the multi-head self-attention (MHSA) attention mechanism and redesigning the network structure to focus on dense small object areas, which can reduce the missed detection and false detection rates of small and occluded objects

Unlike the SPPF structure, which uses different 5×5 maximum pooling to extract high-level and low-level semantic features more effectively, this study uses dilated convolution after dilated convolution. This is more effective than simply using 5×5 maximum pooling to increase the receptive field, and the expression ability of feature maps will be enriched, which is beneficial for object detection tasks with significant differences in object size. T-SPPF first uses the standard CBL module to halve the input channel and then performs a dilated convolution (dilated rate = 2). After the dilated convolution, it performs dilated convolution (dilated rate = 4) and dilated convolution (dilated rate = 6), respectively. Finally, the results of three dilated convolution operations are concatenated with the data without dilated convolution operations in the channel dimension to perform a CBL operation. At the same time, the MHSA [36] module is added to enhance the global information aggregation ability, enabling the model to detect better and recognize small and occluded objects.

3.3. The FP-C2f module

As the number of network layers deepens, the number of feature map channels often increases. Feature maps from multiple channels may carry similar or identical information, which can cause redundancy of feature maps. Therefore, the latest lightweight backbone network, namely FasterNet [37], proposes an idea of partial convolution, which only performs ordinary convolution on some channels of all input feature maps, while the remaining channels that have not been convolved will be saved for subsequent processing. This lightweight method not only reduces computational complexity but also improves the speed of floating-point operations, and because it takes into account the information of all channels, it does not cause serious accuracy degradation. This study is inspired by the partial convolution idea in FasterNet mentioned above and designs the M-Bottleneck structure to replace the Bottleneck structure in the C2f module, thereby reducing the computational complexity of the model. Among them, 1×1 convolution is added to branch 1, firstly because 1×1 convolution can reduce the dimensionality of feature map information, reducing the parameter quantity and computation. Secondly, the 1×1 convolution used in the CBS module undergoes a Swish activation function processing, deepening the network depth and adding more nonlinearity. In the FP-C2f module, 1×3 convolutions and 3×1 convolutions were added, which are asymmetric convolutions [38]. The asymmetric convolution structure formed through this cascading method has the same receptive field as the standard 3×3 convolution. On the one hand, it deepens the depth of the network and enhances the ability of the network nonlinear expression, and different forms of convolution kernels can expand the feature space and better adapt to object features with different width and height ratios. On the other hand, it can save about 33% in terms of parameter quantity.

However, although asymmetric convolution significantly reduces the overall parameter quantity of the network model, it leads to a decrease in the ability of the network model to capture complex features. Therefore, 3×3 PConv modules are added to M-BottleNeck. This increases the width of the model to increase the nonlinear expression ability of the model to extract more feature information from different levels of feature maps in the sample to compensate for the accuracy loss caused by reduced parameter quantity and preserve the original channel information to the greatest extent possible. At the same time, in order to capture the global correlation of feature maps and represent richer connections between semantic features, this study introduces the MHSA module in the M-Bottleneck module. The MHSA module can map input feature sequences to multiple subspaces and perform attention calculations on each subspace, thereby improving the ability to express the object detection model. The MHSA module can also perform different attention calculations on input feature sequences, thereby improving the generalization ability of the object detection model. At the same time, the multi-head attention mechanism can perform more detailed processing on the input feature sequence, thereby improving the accuracy of the object detection model. The network structure of FP-C2f is shown in Fig. 8.

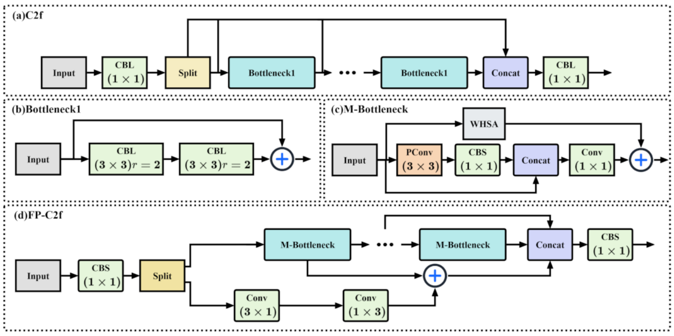


Fig. 8. (a) The network structure of the C2f module. (b) The network structure of the Bottleneck1 module. (c) The network structure of the MHSA-Bottleneck (M-Bottleneck) module. (d) The network structure of the FP-C2f module

3.4. The improvement of overall structure of the LEDF-YOLO network model

When the size of traffic objects is small, the number of pixels is often relatively few. As the number of convolutional layers increases, the collection and processing of semantic feature information by convolutional operations will gradually deepen, resulting in useful features of small traffic objects being easily ignored by network models in convolutional neural networks. This is the fundamental reason why small objects are difficult to detect accurately in object detection tasks. In order to integrate the semantic features of deep and shallow convolutional layers and improve the difficulty of multi-scale traffic object detection, this study concatenates the shallow feature maps in the Backbone section with the deep feature maps of the same size in the PAN structure so that the deep feature maps contain the feature information of the shallow feature maps, as shown by the blue full line in Fig. 9. The low-level semantic information, such as edges and contours of small-scale traffic objects, were incorporated into the deep network structure with a slight increase in parameter size. The final experimental results show that the accuracy of the object detection algorithm has been improved to a certain extent (see Table 4).

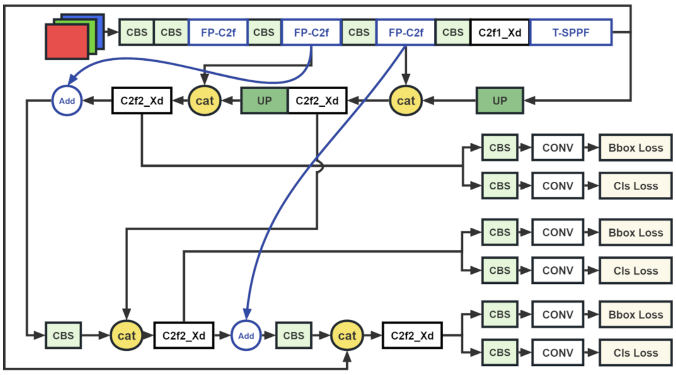


Fig. 9. The network structure of the LEDF-YOLO network model. The blue boxes, circles, and arrows represent the improvements made to the YOLOv8n algorithm in this study

Table 4

The comparison of experimental accuracy on the Microsoft COCO dataset [22] between the improvement network structure and the YOLOv8n algorithm is added. The optimal value for each of these evaluation criteria has been bolded in black, "↑" indicates that a larger value of the evaluation criterion is better, while "↓" indicates that a smaller value of the evaluation criterion is better

Algorithm	mAP@0.5(%)↑	Parameters(M)↓
YOLOv8n	56.4	**3.2**
YOLOv8n with the improvement network structure	**59.7**	3.7

3.5. The construction of the TLLN dataset

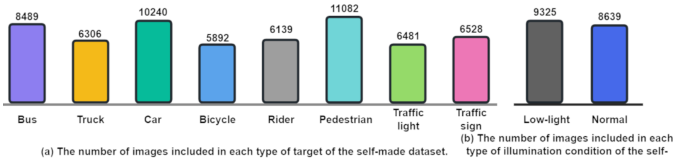
The image of our dataset was mainly captured on roads in Guangzhou, Dalian, and Benxi, containing various roads, various illumination intensities, and different weather conditions. The other part of our dataset was an image formed by shooting the video and then getting frames. A total of 3300 images of the dataset were obtained from video frames. Image size is 1920 × 1080 and 1366 × 768. In addition, 800 images were carefully selected from the ExDark [15] dataset and the BIT-Vehicle [39] dataset. By adding more images to the training set and expanding the sample space to enrich the training set, the data expansion approach may significantly reduce the overfitting of the object detection model and, hence, enhance its generalizability.

The collection scenario of a dataset has an essential impact on the performance of traffic object detection algorithms based on DL. Generally, the higher the quality of the dataset, the higher the accuracy of the object detection algorithm. Therefore, we consider adding images that reflect the traffic status of complex scenes to improve the detection effect and enhance the robustness and adaptability of the model. Meanwhile, in order to balance various samples, various frames containing traffic signs are selected from the captured videos and placed in the dataset. The object detection algorithm detects the objects in the image, labels them with categories, and identifies their positions. We used the LabelMe tool to label the collected images, directly generate corresponding TXT files, and expand the dataset to obtain an object detection dataset including 17 964 images, namely TLLN-Datest, by rotation operation, blur operation, crop operation, and so on. There is a ground-truth annotation document. Among them, each line of the annotation document contains the location information and category of the traffic object in the image. The location information consists of the centre position of the annotation box, as well as the width and height of the box. Before training, convert the tag file to the YOLO format required for training. The dataset instance is shown in Fig. 10.

Fig. 10. The self-made traffic object dataset and examples of traffic objects

4. OBJECT DETECTION MODEL TRAINING

The self-made traffic object dataset and the UA-DETRAC dataset were used for training and test verification on a desktop computer equipped with an NVIDIA GeForce RTX 3070 Ti to validate the practicability and effectiveness of the designed approach. Meanwhile, all object detection algorithms used in this study are without pre-training. The experimental environment runs on Ubuntu 20.04, and the approach proposed in this study is built with the PyTorch framework. During network training, the batch size is 16, the optimizer is Adam, the maximum iteration epoch is 200, the initial learning rate is 0.01, the weight



(a) The number of images included in each type of target of the self-made dataset.

(b) The number of images included in each type of illumination condition of the self-made dataset.

Fig. 11. The number of images included in each type of object of the TLLN dataset and the number of images included in each type of illumination condition of the TLLN dataset

Table 5
Experimental software version and hardware configuration

Name	Model and version
Integrated development environment	PyCharm Community Edition 2022.1.2
GPU-accelerated library of primitives for deep neural networks	CUDA Deep Neural Network 8.4
Graphics card	NVIDIA GeForce RTX 3070Ti
Memory capacity	32GB
Computer processor	Intel i7-13700K Processor
Computer vision and machine learning software library	OpenCV 4.6.0
Deep learning software packages	PyTorch 1.5.1

Table 6
Experimental hyperparameter setting

Hyperparameters	Batch size	Initial learning rate	Epochs	Optimizer
Numerical value	16	0.01	100	Adam
Hyperparameters	Momentum factor	Attenuation coefficient	Warmup epochs	Mosaic data augmentation probability
Numerical value	0.937	0.0005	3	1.0

attenuation coefficient is 0.0005, and the momentum is 0.937. The software version and hardware configuration of this network model are shown in Table 5. The hyperparameter settings of this study are shown in Table 6.

4.1. Evaluation indicators

The full name of the average precision (AP) indicator is the average precision, which is an essential indicator for measuring the detection accuracy of a single object category. Before introducing AP metrics, first introduce two key concepts: Precision and Recall. The Precision and Recall are calculated separately for a specific category, and the formula for calculating Precision and Recall is as follows:

$$Precision = \frac{TP}{TP+FP} \times 100\%, \tag{3}$$

$$Recall = \frac{TP}{TP+FN} \times 100\%, \tag{4}$$

where TP denotes a correctly classified object with intersection over union (IoU) against the truth box higher than the threshold, FP means that the predicted bounding box has IoU less than the threshold, or the object does not exist, and the model detects one. And finally, FN denotes the ground truth objects that have no prediction.

The AP is defined by:

$$AP = \sum_{n} (R_n - R_{n-1}) P_n, \tag{5}$$

where R_n and P_n are the precision and recall at the nth threshold.

The mean average precision (mAP) indicator represents the average of different types of average precision. The mAP is defined by:

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i, \tag{6}$$

where N the number of detected categories, i denotes the detected category.

The mAP@0.5 indicator represents the probability of accurate prediction when the IoU of the candidate bound and ground truth bound is greater than 0.5.

4.2. Comparison with related algorithms

To further validate the detection performance of the LEDF-YOLO method proposed in this study, we will compare the proposed LEDF-YOLO method model with mainstream lightweight object detection methods such as YOLOv5n and YOLOv8n in this section. The detection results of each detection algorithm on the self-made dataset are shown in Table 7 and Fig. 12. Figure 12 shows the curves of precision, recall, and mAP@0.5 of the YOLOv8n and LEDF-YOLO method. By analyzing Table 7 and Fig. 12, it can be seen that the LEDF-YOLO model proposed in this study has significant advantages in the mAP@0.5 indicator, precision indicator, and recall indicator. The detection

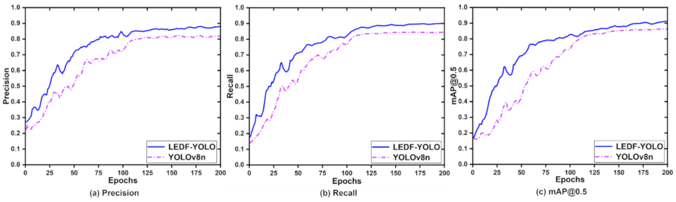


Fig. 12. The curves of the four evaluation indicators for the LEDF-YOLO algorithm and the YOLOv8n algorithm during training on the self-made dataset. (a) The precision curve. (b) The recall curve. (c) The mAP@0.5 curve

Table 7

Comparison with mainstream object detection approaches on the self-made dataset (The best results are bolded). The optimal value for each of these evaluation criteria has been bolded in black, "↑" indicates that a larger value of the evaluation criterion is better, while "↓" indicates that a smaller value of the evaluation criterion is better

Model	Size	Precision(%)↑	Recall(%)↑	mAP(%)↑	FPS(frame/s)↑
YOLOv5n	640	79.63	80.56	82.30	**129.1**
YOLOv8n	640	81.97	83.72	85.41	126.9
LEDF-YOLO	640	**89.52**	**90.48**	**92.75**	121.4

accuracy of LEDF-YOLO is higher than that of other detection models. The overall real-time difference between LEDF-YOLO and YOLOv8n is small. However, LEDF-YOLO has a precision indicator and recall indicator that is 7.55 % and 6.76 % higher than YOLOv8n, respectively. While YOLOv5n is slightly higher in speed of reasoning than LEDF-YOLO and YOLOv8n, the detection accuracy of YOLOv5n and YOLOv8n is relatively higher compared to YOLOv5n. Therefore, the real-time indicator and detection accuracy of the LEDF-YOLO method meet the requirements of efficient and real-time accurate detection for traffic object detection.

In order to verify that the improved method in this study can effectively improve the detection effect of traffic objects, images from different scenes in the self-made dataset were selected for comparison in the experiment. The detection results are shown in Fig. 7. The car in the bottom right corner of Image 8 and the car on the right side of Image 3, due to image size limitations, only partially appear in the image. Therefore, the original YOLOv8n algorithm cannot accurately detect them. However, the improved method in this study still detects them even when the object is partially missing. As shown in Image 7, in scenes with densely arranged traffic objects, the YOLOv8n algorithm falsely detected a bus as a truck, but the method proposed in this study successfully detected them. As shown in Image 5, due to the small size of the object, interference from background information, and overlapping objects, the YOLOv8n algorithm missed detecting pedestrians on the roadside. However, our proposed improved method was able to detect this small object. As shown in Image 1, in a dark night scene, due to insufficient lighting, the edges and details of objects become blurred. Therefore, the YOLOv8n algorithm missed and misclassified some traffic signs and vehicles in the image, while our proposed improved method successfully detected them. These results demonstrate that the improved method proposed in this study improves the accuracy of traffic object detection in different traffic scenarios.

In order to further compare LEDF-YOLO with other detection models, this work conducted comparison tests on the UA-DETRAC dataset to validate the generalization performance of the LEDF-YOLO algorithm. The comparison testing findings demonstrate that the LEDF-YOLO algorithm has much superior detection accuracy indicators than other methods, as shown in Table 8, and the detection speed has reached 131 FPS. This implies that the LEDF-YOLO algorithm has high detection performance and can swiftly identify and locate traffic objects, which is suitable for practical implementation. Overall, there is no significant difference between the method described in this study and the YOLOv8n algorithms in terms of parameter amount and FPS indicators. This demonstrates that the enhancement of the LEDF-YOLO algorithm over the YOLOv8n algorithm can effectively improve object detection accuracy while meeting real-time requirements, making the model suitable for deploying in autonomous driving systems that require high real-time and object detection accuracy while meeting the needs of vehicle intelligent development.

Table 8

Comparison between the mainstream object detection approaches on the UA-DETRAC [9] dataset. From the table, it can be seen that compared to the baseline algorithm, YOLOv8n, LEDF-YOLO significantly improves the accuracy of object detection while ensuring real-time performance. The optimal value for each of these evaluation criteria has been bolded in black, "↑" indicates that a larger value of the evaluation criterion is better, while "↓" indicates that a smaller value of the evaluation criterion is better

Algorithm	Backbone	Size	FLOPs(G)↓	Params(M)↓	mAP(%)↑	FPS(frame/s)↑
SSD [20]	VGG-Net	300	31.7	26.3	73.23	53
EfficientDet	EfficientNets	512	17.9	17.1	68.39	29
YOLOv7	E-ELAN-based	640	104.7	37.4	97.89	64
YOLOX-tiny	CSPDarknet53	640	**6.4**	5.1	87.25	61
YOLO7-tiny	E-ELAN-based	640	6.5	6.2	83.13	116
YOLOv8n	C2f-based	640	8.7	3.5	90.71	107
LLD-YOLO [40]	C3k2f-based	640	25.1	11.2	92.81	95
YOLOv12n [41] with DD-GAN	R-ELAN-based	640	7.6	**3.3**	94.93	**128**
LEDF-YOLO	FP-C2f-based	640	9.5	4.2	**98.13**	101

5. ABLATION STUDY

In order to evaluate the effectiveness of various improvement strategies adopted in this study on the LEDF-YOLO algorithm, ablation experiments were conducted in the self-made dataset constructed in this study to analyze the performance effects of different improvement strategies quantitatively. The results of ablation experiments are shown in Table 9.

In Table 9, the Group 1 experiment was directly conducted using YOLOv8n. The experimental results show that the mAP@0.5 indicator of YOLOv7 reached 85.41%, and the detection speed is 126.9 FPS. Through experimental results, it can be seen that compared with existing traffic object detection methods, the YOLOv8n algorithm has advantages in speed, but its accuracy is still not ideal. In the Group 2 experiment, the image-enhancement module was introduced into the YOLOv8n algorithm, and the mAP@0.5 indicator reached 89.18%, an increase of 3.77% compared to the YOLOv8n algorithm. This proves the effectiveness of adding the image-enhancement module strategy to the YOLOv8n algorithm. The Group 3 experiment introduced the FP-C2f module into the Group 2 network model, and the mAP@0.5 indicator increased by 3.70% while FPS only decreased by 1.6%. This indicates that the FP-C2f module can effectively improve the accuracy of traffic object detection. Group 4 introduced the FP-C2f module into the Group 3 network model. At this time, the mAP@0.5 indicator reached 92.47%. Compared with the original YOLOv8n algorithm, the mAP@0.5 indicator increased by 7.06%, and the FPS indicator of the network model did not significantly decrease. This indicates that the FP-C2f module maintains real-time while better-capturing image features through attention mechanisms and new network structures, enhancing the expression ability of the network model and making it more suitable for application in edge devices. The Group 5 experiment is a combination of all improvement points, with the mAP@0.5 indicator of 92.75% and the detection speed of 121.4 FPS. Although the detection speed of the TOD-YOLOv7 algorithm has slightly decreased, the mAP@0.5 indicator has increased by 7.34%. Therefore, the LEDF-YOLO algorithm has a good detection effect.

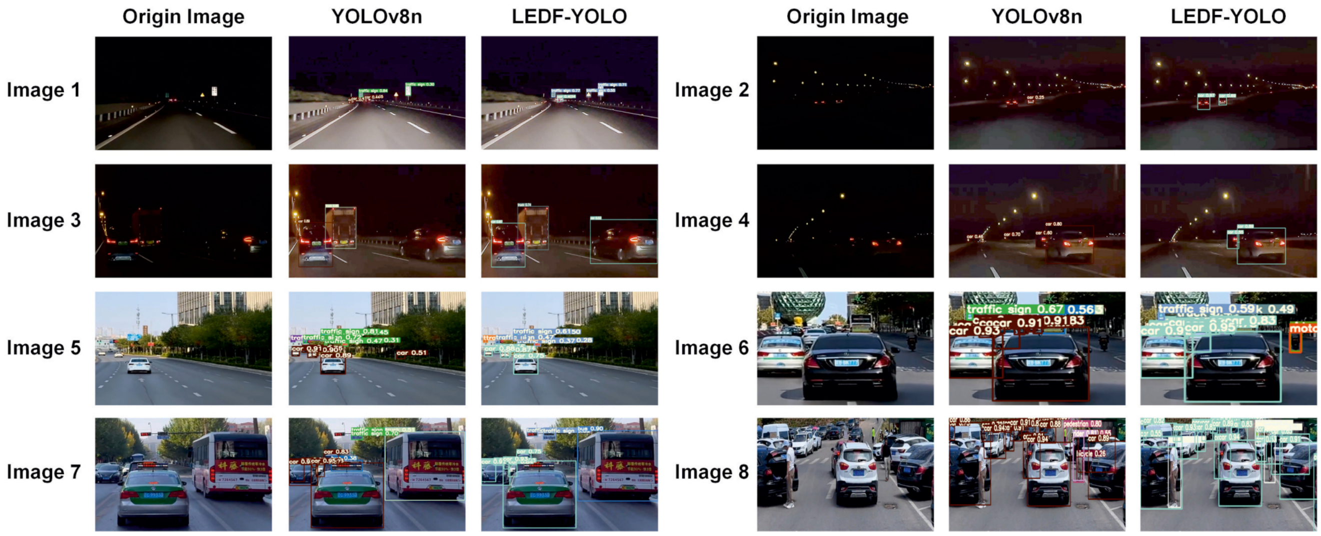


Fig. 13. Visual comparison with relevant methods on the self-made dataset (the images detected using the YOLOv8n algorithm are also enhanced using the DD-GAN algorithm)

Table 9

Comparison between related detection algorithms on the self-made dataset. It can be seen that after adding the low-illumination image enhancement algorithm (DD-GAN) to the preprocessing step of the object detection algorithm, the detection accuracy of the object detection algorithm has been greatly improved without affecting the inference speed too much. Among them, "✓" means adding the corresponding module, "×" means not adding the corresponding module (the best results are bolded). Overall, the LEDF-YOLO algorithm improves the detection accuracy of the algorithm, effectively balancing accuracy and real-time

Group	Module		Improvement of network structure	DD-GAN	mAP@0.5(%)	FPS(frame/s)
	FP-C2f	T-SPPF				
1 (YOLOv8n)	×	×	×	×	85.41	**126.9**
2	×	×	×	✓	89.18	124.7
3	✓	×	×	✓	92.89	123.1
4	✓	✓	×	✓	92.47	121.9
5 (LEDF-YOLO)	✓	✓	✓	✓	**92.75**	121.4

6. CONCLUSIONS

This study proposes a lightweight low-illumination traffic object detection method, namely LEDF-YOLO, based on image enhancement to solve the problem of poor detection effect of object detection algorithms for traffic objects in low-illumination traffic scenes. Firstly, the SPPF module and C2f module of the YOLOv8n algorithm have been optimized and improved to extract and fuse important semantic feature information of traffic objects. Secondly, an efficient DD-GAN network module is proposed to enable the object detection algorithm to obtain richer semantic information. In addition, the network model of the YOLOv8n algorithm has been improved, allowing it to pay more attention to the semantic feature information of small and occluded objects and effectively process object features of different scales, thereby effectively reducing the occurrence of missed and false detections. Validation was conducted on the UA-DETRAC dataset and self-made dataset, and the experimental results showed that the Precision indicator of the LEDF-YOLO algorithm is improved by 7.55% when compared to the YOLOv8n algorithm, the Recall indicator is improved by 6.76%, the mAP@0.5 indicator is improved by 7.34%, and the detection speed reaches 121.4 FPS, which realizes the real-time and high-precision detection of traffic objects.

With the development of AI, the performance of traffic object detection algorithms is gradually improving, but many problems still need to be solved urgently. The background of traffic images is relatively complex, and this study only focuses on image enhancement under low-illumination conditions. Adverse environments, such as foggy days and sandstorms, can also affect the effectiveness of object detection algorithms in detecting traffic objects. Therefore, image enhancement for different weather environments will be a further research direction.

ACKNOWLEDGEMENTS

This work was supported by the Qinglan Project of Jiangsu Province of China. The authors conducted this work with support from our cooperative companies.

REFERENCES

[1] K. Hanzel, K. Paszek, and D. Grzechca, "The influence of the data packet size on positioning parameters of uwb system for the purpose of tagging smart city infrastructure," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 68, no. 4, pp. 857–868, 2020, doi: 10.24425/bpasts.2020.134173.

[2] B. Paprocki, A. Pregowska, and J. Szczepanski, "Optimizing information processing in brain-inspired neural networks," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 68, no. 2, pp. 225–233, 2020, doi: 10.24425/BPASTS.2020.131844.

[3] S. Kasarapu and S.M.P. Dinakarrao, "Performance and environment-aware advanced driving assistance systems," *IEEE Trans. Comput.*, vol. 74, no. 1, pp. 231–242, 2024, doi: 10.1109/TC.2024.3475572.

[4] R. Kapela, "Texture recognition system based on the deep neural network," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 68, no. 6, pp. 1503–1511, 2020, doi: 10.24425/BPASTS.2020.135395.

[5] Y. Hu *et al.*, "Dynamic center point learning for multiple object tracking under severe occlusions," *Knowledge-Based Syst.*, vol. 300, p. 112130, 2024, doi: 10.1016/j.knosys.2024.112130.

[6] A.M. Roy, R. Bose, and J. Bhaduri, "A fast accurate fine-grain object detection model based on yolov4 deep neural network," *Neural Comput. Appl.*, vol. 34, no. 5, pp. 3895–3921, 2022, doi: 10.1007/s00521-021-06651-x.

[7] G. Singal, H. Singhal, R. Kushwaha, V. Veeramsetty, T. Badal, and S. Lamba, "Roadway: lane detection for autonomous driving vehicles via deep learning," *Multimed. Tools Appl.*, vol. 82, no. 4, pp. 4965–4978, 2023, doi: 10.1007/s11042-022-12171-0.

[8] Q. Yan *et al.*, "Uncertainty estimation in HDR imaging with Bayesian neural networks," *Pattern Recognit.*, vol. 156, p. 110802, 2024, doi: 10.1016/j.patcog.2024.110802.

[9] L. Wen *et al.*, "Ua-detrac: A new benchmark and protocol for multi-object detection and tracking," *Comput. Vis. Image Underst.*, vol. 193, p. 102907, 2020, doi: 10.1016/ J.CVIU. 2020. 102907.

[10] S. Plainis and I. Murray, "Reaction times as an index of visual conspicuity when driving at night," *Ophthalmic Physiol. Opt.*, vol. 22, no. 5, pp. 409–415, 2002, doi: 10.1046/j.1475-1313.2002.00076.x.

[11] R. G. Baker, "On the quantum mechanics of optic flow and its application to driving in uncertain environments," *Transp. Res. Pt. F-Traffic Psychol. Behav.*, vol. 2, no. 1, pp. 27–53, 1999, doi: 10.1016/S1369-8478(99)00005-4.

[12] P. Cong, H. Feng, S. Li, T. Li, Y. Xu, and X. Zhang, "A visual detection algorithm for autonomous driving road environment perception," *Eng. Appl. Artif. Intell.*, vol. 133, p. 108034, 2024, doi: 10.1016/j.engappai.2024.108034.

[13] H. Gao, J. Shao, M. Iqbal, Y. Wang, and Z. Xiang, "Cfpc: The curbed fake point collector to pseudo-lidar-based 3d object detection for autonomous vehicles," *IEEE Trans. Veh. Technol.*, vol. 72, no. 2, pp. 1922–1934, 2024, doi: 10.1109/TVT.2024.3372940.

[14] T. Wang, H. Qu, C. Liu, T. Zheng, and Z. Lyu, "Lle-std: Traffic sign detection method based on low-light image enhancement and small target detection," *Mathematics*, vol. 12, no. 19, p. 3125, 2024, doi: 10.3390/math12193125.

[15] Y. P. Loh and C. S. Chan, "Getting to know low-light images with the exclusively dark dataset," *Comput. Vis. Image Underst.*, vol. 178, pp. 30–42, 2019, doi: 10.1016/j.cviu.2018.10.010.

[16] Y. Sasagawa and H. Nagahara, "Yolo in the dark-domain adaptation method for merging multiple models," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*. Springer, 2020, pp. 345–359, doi: 10.1007/978-3-030-58589-1_21.

[17] C. Chen, Q. Chen, J. Xu, and V. Koltun, "Learning to see in the dark," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3291–3300, doi: 10.1109/CVPR.2018.00347.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788, doi: 10.1109/cvpr.2016.91.

[19] W. Chen and T. Shah, "Exploring low-light object detection techniques," *arXiv preprint arXiv:2107.14382*, 2021, doi: 10.48550/arXiv.2107.14382.

[20] W. Liu *et al.*, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amster-*

dam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.

[21] I. Goodfellow *et al.*, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020, doi: 10.1016/b978-0-32-396126-4.00015-1.

[22] T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 2014, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.

[23] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, pp. 303–338, 2010, doi: 10.1007/s11263-009-0275-4.

[24] C. Li *et al.*, "Yolov6: A single-stage object detection framework for industrial applications," *arXiv preprint arXiv:2209.02976*, 2022, doi: 10.48550/arXiv.2209.02976.

[25] 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. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7464–7475, doi: 10.48550/arXiv.2207.02696.

[26] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020, doi: 10.48550/arXiv.2004.10934.

[27] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021, doi: 10.48550/arXiv.2107.08430.

[28] Y. Jiang *et al.*, "Enlightengan: Deep light enhancement without paired supervision," *IEEE Trans. Image Process.*, vol. 30, pp. 2340–2349, 2021, doi: 10.1109/TIP.2021.3051462.

[29] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proc. European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19, doi: 10.1007/978-3-030-01234-2_1.

[30] C. Lee, C. Lee, and C.-S. Kim, "Contrast enhancement based on layered difference representation," in *2012 19th IEEE International Conference on Image Processing*. IEEE, 2012, pp. 965–968, doi: 10.1109/ICIP.2012.6467022.

[31] K. Ma, K. Zeng, and Z. Wang, "Perceptual quality assessment for multi-exposure image fusion," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3345–3356, 2015, doi: 10.1109/TIP.2015.2442920.

[32] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015, doi: 10.48550/arXiv:1511.07122.

[33] C. Guo *et al.*, "Zero-reference deep curve estimation for low-light image enhancement," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1780–1789, doi: 10.1109/CVPR42600.2020.00185.

[34] F. Zhang, Y. Li, S. You, and Y. Fu, "Learning temporal consistency for low light video enhancement from single images," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4967–4976, doi: 10.1109/CVPR46437.2021.00493.

[35] Y. Zhang, J. Zhang, and X. Guo, "Kindling the darkness: A practical low-light image enhancer," in *Proc. 27th ACM International Conference on Multimedia*, 2019, pp. 1632–1640, doi: 10.1145/3343031.3350926.

[36] A. Vaswani *et al.*, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 30, pp. 5998–6008, 2017, doi: 10.48550/arXiv.1706.03762.

[37] J. Chen *et al.*, "Run, don't walk: Chasing higher flops for faster neural networks," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 021–12 031, doi: 10.1109/CVPR52729.2023.01157.

[38] X. Ding, Y. Guo, G. Ding, and J. Han, "Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks," in *Proc. IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1911–1920, doi: 10.1109/ICCV.2019.00200.

[39] Z. Dong, Y. Wu, M. Pei, and Y. Jia, "Vehicle type classification using a semisupervised convolutional neural network," *EEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2247–2256, 2015, doi: 10.1109/TITS.2015.2402438.

[40] Q. Zhang, W. Guo, and M. Lin, "Lld-yolo: a multi-module network for robust vehicle detection in low-light conditions," *Signal Image Video Process.*, vol. 19, no. 4, pp. 1–11, 2025, doi: 10.1007/s11760-025-03858-6.

[41] Y. Tian, Q. Ye, and D. Doermann, "Yolov12: Attention-centric real-time object detectors," *arXiv preprint arXiv:2502.12524*, 2025, doi: 10.48550/arXiv.2502.12524.