Applied Computing and Intelligence

https://www.aimspress.com/journal/aci

Applied Computing and Intelligence
6(1): 23–37.
DOI: 10.3934/aci.2026002
Received: 07 November 2025
Revised: 23 January 2026
Accepted: 28 January 2026
Published: 04 February 2026

Research article

Nighttime vehicle target detection based on visual features

Ling Peng and Libiao Jiang*

School of Mechanical Engineering, Guangzhou City University of Technology, No. 1 Xuefu Road, Huadu District, Guangzhou City, Guangdong Province, China

* **Correspondence:** Email: pengl@gcu.edu.cn; Tel +8618620716791.

Academic Editor: Jidong Yang

Abstract: This study proposes an algorithm based on an improved YOLOv8s for nighttime vehicle recognition under low light and complex background conditions, addressing issues such as shallow feature loss, low recognition accuracy for occluded targets, and bounding box distortion. The core optimization strategies are as follows: First, RepGFPN (Re-parameterized Generalized Feature Pyramid Network) employs re-parameterized fusion, utilizing a customized fusion module to achieve multi-scale feature interaction. This preserves shallow-level details while eliminating redundant sampling and adjusting channels to enhance fusion efficiency. Second, the DAT (Deformable Attention Transformer) attention mechanism generates reference grids, while the Lightweight Network predicts deformation point offsets. Bilinear interpolation extracts key features, enhancing learning capabilities for small-sized and occluded targets. Third, the MPDIoU (Minimum Point Distance Intersection over Union) loss function minimizes Euclidean distances between bounding box corners, integrating intersection-over-union optimization to mitigate deformation caused by vehicle overlap. Experimental results demonstrate superior performance over the original model: accuracy improved by 6.4%, recall increased by 4.9%, latency reduced by 6.3 ms, mAP0.5 was boosted by 4.4%, and N_Params decreased by 2.5.

Keywords: autonomous vehicles; visual features; vehicle detection; algorithm optimization

1. Introduction

The use of big data and artificial intelligence has significantly advanced autonomous driving technology in recent years. Improved and optimized advanced driver assistance systems and mature autonomous driving technology are important ways to address road traffic safety issues and successfully reduce traffic accidents. The three primary services that assist systems offer are vehicle

recognition, tracking, and behavior prediction, which are essential in traffic scenarios. The key link for environmental perception is effective vehicle detection and recognition. This essential traffic aspect can dramatically increase road driving safety by expanding the perceptual range of the driver and accelerating reaction and decision-making. Autonomous navigation forms the core foundation of intelligent driving, centered on dynamically planning safe routes through real-time environmental perception. Nighttime vehicle detection serves as the critical pillar of this perception process, enabling precise capture of vehicle target information in nighttime driving scenarios. Only by deeply integrating the precise positioning required for autonomous navigation with nighttime vehicle detection can a reliable basis be provided for dynamic path adjustments, effectively mitigating obstacle risks. Furthermore, driver assistance systems heavily rely on nighttime vehicle detection technology to compensate for the driver's diminished attention and delayed reactions in low-light conditions, thereby ensuring driving safety.

Although daytime vehicle detection technology has advanced and become extensively employed, nighttime vehicle detection is a technical obstacle that needs to be overcome. Vehicle shape, color, texture, and other visual characteristics are blurred or even difficult to record due to the effects of dark lighting. The lack of complex scene datasets and the coupled effects of extreme weather conditions lead to insufficient model generalization capabilities. These inadequacies increase the difficulty for nighttime vehicle target detection. Thus, enhancing vehicle detection technology at night contributes to raising the general level of traffic safety.

At present, the most common approaches for vehicle target detection are deep learning and classical feature-driven methods. Traditional approaches have poor adaptability, subjectivity in feature selection, and low computing efficiency. Thus, they are unsuitable for complex and dynamic traffic scenarios and are gradually being phased out. Target detection requires extensive use of deep learning, which has become the dominant paradigm for vehicle object detection, gradually replacing traditional feature-driven approaches. The latter suffer from poor environmental adaptability, strong subjectivity in manual feature selection, and low computational efficiency—limitations that make them ill-suited for complex, dynamic traffic scenarios. Deep learning–based approaches can be categorized into single-stage regression algorithms [e.g., Single Shot Multibox Detector (SSD) [1] and You Only Look Once (YOLO) series [2]] and two-stage detection algorithms based on candidate areas [e.g., Region-based Convolutional Neural Network (RCNN) [3], Spatial Pyramid Pooling Network (SPP-NET) [4], and Fast Region-based Convolutional Neural Network (Fast-RCNN) [5]]. The latter uses an end-to-end design and directly outputs the object category and location information to obtain a faster detection response. Meanwhile, the former first generates candidate regions and then fine-classifies them to ensure high accuracy at the sacrifice of speed. Each of these algorithms has innate advantages, and their combinations propel the ongoing advancement of autonomous driving technology in vehicle identification. The iterative development and cross-integration of these methods continue to advance the application of autonomous driving technology in vehicle recognition. However, nighttime vehicle detection still faces unresolved core challenges.

Notably, despite multi-dimensional explorations of optimization pathways, existing research remains constrained by limitations. Authors in [6] proposed an SNN model integrating locally modulated IF neurons with DCNet, balancing accuracy and low power consumption, but suffering from post-conversion accuracy loss. In [7], authors enhanced small traffic sign detection using DCSP and IENet, adapting to complex environments yet struggling with extreme noise. The work in [8] employs a two-stage metadata optimization to bridge the gap between virtual and real-world scenarios, but the model is overly complex. In [9], a multi-granularity adaptive nighttime detection system was developed, yet the complexity of its backbone network limits its adaptability to image distortion and

glare.

In vehicle detection, [10] proposed E-YOLOv7 for nighttime vehicle detection, incorporating an SFE feature enhancement module, GS-EFF lightweight fusion network, and Soft-EIoU-NMS post-processing to significantly improve detection performance for small/dense targets. However, this approach focuses solely on small targets, lacks adaptability to extreme weather conditions, and does not integrate non-visual information. In traffic scene detection, [11] employed YOLOv8 on the ZND dataset for traffic sign detection, revealing a significant correlation between reflectivity and nighttime detection performance. However, this approach targets traffic signs rather than vehicles, relies on physical properties, and fails to address vehicle-specific challenges like severe occlusion.

In contrast, this study effectively addresses three core challenges in nighttime vehicle detection—shallow feature loss, low recognition accuracy for occluded targets, and bounding box distortion—by optimizing YOLOv8s, thereby overcoming limitations of existing methods. Specifically, the Deformable Attention Transformer (DAT) enhances the representation of small/occluded objects, RepGFPN employs bidirectional fusion to preserve weak low-light features, and MPDIoU loss mitigates bounding box distortion caused by vehicle overlap. This approach achieves an optimized balance between detection accuracy, speed, and adaptability to complex nighttime scenarios, providing a more robust solution for autonomous driving's nighttime environmental perception.

2. Algorithm for nighttime vehicle detection

2.1. Overview of YOLOV8S

The YOLO series target detection algorithms are extensively utilized in real-time detection due to their great comprehensive performance and quick detection speed. YOLOv8s is a new version of the YOLO series of target detection algorithms. It retains the benefits of the original method while adding certain enhancements to boost speed. The detection algorithm can operate on various software platforms due to the new backbone network, the Ancher-Free detection header and the redesigned loss function, which results in a greater range of application scenarios and increased efficiency. In this study, YOLOv8s is selected as the benchmark network for algorithm optimization to minimize the risk of misdetection and omission and guarantee that the real-time requirements are fulfilled. This optimization fully utilizes the lightweight network structure of YOLOv8s and further refines it for the challenging points of nighttime vehicle detection.

2.2. Algorithm optimization

2.2.1. Design of feature fusion network

The low accuracy of nighttime small-scale vehicle target recognition is primarily attributed to the loss of some shallow small-scale feature information caused by the rise in network depth. Semantic information at various levels is usually integrated using a feature fusion network. The structure of the Feature Pyramid Network (FPN) is established. After a top-down path is created, feature fusion is used to balance the high semantics of higher-level features with the high resolution of low-level features, which results in high robustness. However, the improvement in accuracy is limited by the single structure of the FPN. The FPN + PAN structure increases the accuracy of target recognition by first adding a bottom-up path. Then, the position information is sent to the feature map via this path, which provides it with rich position information and high semantic information. However, both strategies ignore the internal connections of the network and are single-item structures. The structure of the

feature fusion-based GFPN is also established. This structure integrates the information exchange of various connection layers and combines the various scale properties of the previous and current levels to fully share high-level semantic information and low-level spatial information [12–14]. However, the GFPN encounters a problem with growing inference time, which poses difficulties in balancing the speed and detection accuracy of the model.

Thus, this study uses a unique feature fusion approach [15,16] for RepGFPN to address the aforementioned issues. First, the fusion block module is developed to perform feature fusion for increasing detection accuracy based on the same computational complexity. Figure 1 illustrates the structure of the fusion block module. Second, multiple sampling operations are eliminated during the feature fusion stage to increase detection efficiency without sacrificing accuracy. Lastly, different feature levels adopt varying numbers of channels to regulate the feature expression ability.

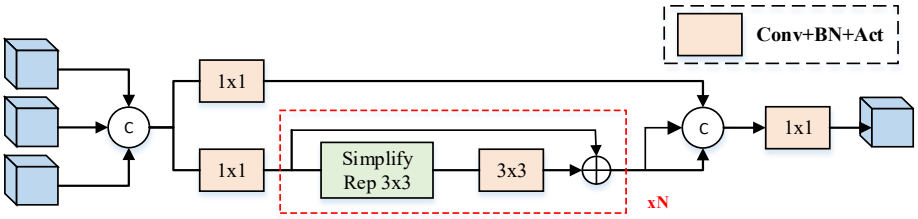

Figure 1. Structure of the fusion block.

RepGFPN replaces the original FPN in the feature fusion stage of YOLOv8s, with clear integration logic and structural details design as elaborated below:

Module composition: The RepGFPN comprises four RepGFPN blocks, corresponding to the backbone output feature maps C3 (80 × 80 × 256), C4 (40 × 40 × 512), C5 (20 × 20 × 1024), and an additional top-level feature map (10 × 10 × 2048). The top-level feature map is generated by a 3 × 3 convolution on the deepest backbone output to enhance high-level semantic representation.

Operator-level structure: Each block uses a re-parameterizable design. In the training phase, two parallel branches are used in a 1 × 1 convolution for channel adjustment and a 3 × 3 depthwise separable convolution for efficient feature extraction. In the inference phase, the dual branches are fused into a single 3 × 3 convolution via kernel and bias re-parameterization, balancing training flexibility and inference efficiency.

Channel dimensions: During fusion, the channel dimensions of feature maps are strictly maintained: C3 (256 channels), C4 (512 channels), C5 (1024 channels), and the top-level feature map is adjusted to 1024 channels for consistent interaction. Finally, the RepGFPN outputs three-scale detection feature maps (80 × 80 × 256, 40 × 40 × 512, 20 × 20 × 1024), which are fed into the subsequent detection head for target localization and classification.

2.2.2. Attention mechanisms

This study incorporates the attention mechanism into the YOLOv8s detection network to improve the feature learning capability of the model for small-scale and obstructed targets. The integration involves modifying the weights of critical features and decreasing the expression of redundant information. The enhanced network model can better extract the key features of the target to improve the accuracy of target detection. At present, CNN-based and Transformer models are common components of attention mechanism models in computer vision.

Attention mechanisms based on CNN models are a technique introduced in specific layers of CNNs. This approach is applied to particular CNN layers to improve the attention of the model to

particular areas or channels of an image, which involves either adding more attention modules or changing the CNN structure. However, CNNs typically employ a fixed-size convolutional kernel, with each neuron having a limited receptive field. Such a limitation prevents direct access to global information. They also exhibit scale invariance, which makes them useless for objects with significantly varying target scales.

Using the self-attention mechanism, the Transformer model-based attention mechanism applies directly to graphical data to capture the association between various picture regions. Transformer-based models are more appropriate for complex and changeable scenes at night and vehicle targets of different sizes [17]. They can also be applied to feature maps of different scales and are appropriate for remote changing scenes with a large field of view, as well as image data of varying sizes and structures without assuming or restricting the image structure. Popular attention mechanisms based on the Transformer model, such as ViT (Vision Transformer), PVT (Pyramid Vision Transformer), and Swin Transformer, possess wider sensory fields and the ability to focus on various levels and sizes of information simultaneously. However, overfocusing can result in higher computational costs, slower convergence, and an increased risk of overfitting [18].

The mechanism of the Deformable Self-Attention Transformer (DAT) is established as well. By concentrating on the pertinent information, the data determines where key and value pairs should be placed [19]. Figure 2 compares DAT with other Vision Transformer models and DCN (Deformable Convolutional Networks) in the CNN model. The regions included in the query are indicated by the various colored masks, while the red and blue stars denote distinct inquiries. Particularly, the DCN model learns distinct deformation points for every query. The DAT model learns shared deformation points for all queries in a data-dependent manner. The ViT model uses global attention for all queries. Meanwhile, Swin Transformer uses the window-spacing attention technique.

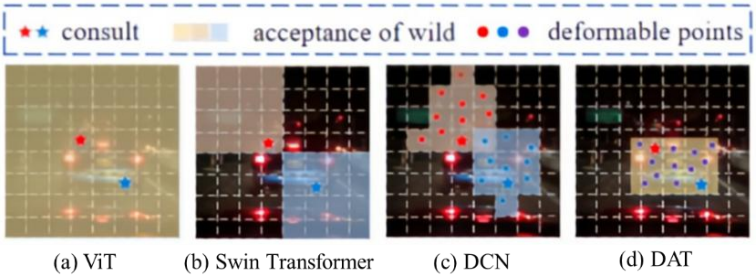

Figure 2. Comparison of different attention models.

Bilinear interpolation feature sampling for feature mapping is the foundation of the DAT model. The distorted content is obtained by feeding the sampling data into key and value projections. A multi-head attention method is employed to concentrate on sampled key questions and aggregate features. The model focuses more on the region of interest and gathers more information as a result of the relative positional bias provided by the positions of the deformed points. This improves the learning of deformable attention. As a result, this approach is especially well-suited for handling dim lighting and hazy visual elements at night. It concentrates more on the vehicle area and the important regions of the target area, which not only minimizes computation but also guarantees recognition accuracy.

The structural principle of DAT is shown in Figure 3. Given the input feature map $x \in R^{H\times W\times C}$, a uniform grid of points $x \in R^{H_G\times W_G\times 2}$ is generated as a reference. Specifically, the grid size is reduced from the input feature map size by a factor r, where $H_G = H/r, W_G = W/r$. The values of the reference point are linearly spaced 2D coordinates. These coordinates are then normalized to the range $[-1, +1]$, where $(-1, -1)$ denotes the upper-left corner, and $(+1, +1)$ denotes the lower-right

corner, based on the mesh shape $H_G \times W_G$. The query token ^{q} is obtained by means of a feature-mapped linear projection to query the offset of each reference point in turn. It is then fed into a lightweight sub-network $\theta_{offset}(.)$ to generate the offsets Δp. The training process is stabilized using a predefined factor s to measure the amplitude of Δp, that is, $\Delta p \leftarrow s\, tanh(\Delta p)$. The features are sampled at the locations of the deformation points as keys and values, and the projection matrix is sampled as follows:

$$q = xW_q, \tilde{k} = \tilde{x}W_k, \tilde{v} = \tilde{x}W_v, \tag{1}$$

$$\Delta p = \theta_{offset(q)}, \tilde{x} = \varphi(x; p + \Delta p), \tag{2}$$

where q is the query token; x is the input feature map; W_q, W_k, and W_v are the projection matrices; Δp is the offset; $\theta_{offset}(.)$ is the offset network; $\tilde{k}$ is the deformation key embedding; $\tilde{v}$ is the value embedding; and $\tilde{x}$ is the deformed point after sampling. Among them, the sampling function $\phi(.;.)$ is set in the form of bilinear interpolation:

$$\varphi(z; (p_x, p_y)) = \sum_{(r_x, r_y)} g(p_x, r_x)\, g(p_y, r_y) z[r_y, r_x, :], \tag{3}$$

where the function $g(a, b) = max(0, 1 - |a - b|)$, (r_y, r_y), indexes all positions of the grid points $z \in R^{H \times W \times C}$, (p_x, p_y). The output of the attention header for the query embedding q, the key embedding k, and the value embedding v is multiplexed with relative positional offsets R. The output of the attention header is formulated as follows:

$$z = Concat\left(z^{(1)}, \ldots, z^{(M)}\right) W_o,$$

$$z^{(m)} = \sigma(q^{(m)} \tilde{k}^{(m)} / \sqrt{d} + \varphi(\hat{B}; R)) \tilde{v}^{(m)}, \tag{4}$$

$$z = Concat(z^{(1)}, \ldots, z^{(m)}) W_o, \tag{5}$$

where $\sigma(.)$ is the softmax function, $z^{(m)}$ represents the embedding output of the ^{m} th attention head, $d = C/M$ is the head dimension, and M represents the number of attention heads in the multi-head self-attention block. C represents the number of channels at each position in the input feature map. $q^{(m)}$, $\tilde{k}^{(m)}$, and $\tilde{v}^{(m)} \in R^{N \times d}$ are the query embedding, key embedding, and value embedding, respectively. $\phi(\hat{B}; R) \in R^{HW \times H_G W_G}$ corresponds to the position embedding. Some adaptations are conducted to connect the features of each head together. The projection is performed with W_o to obtain the final output z.

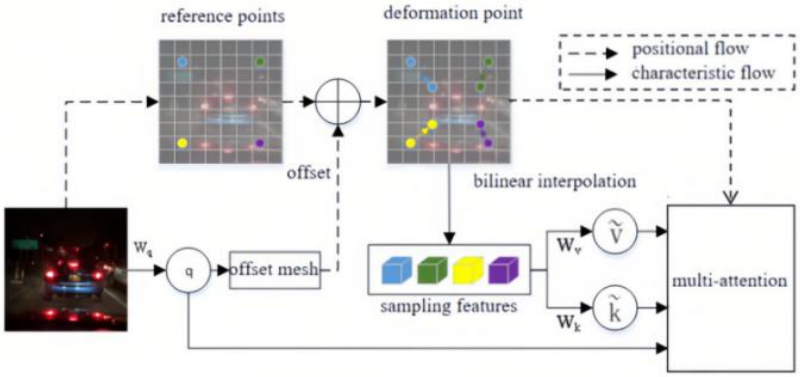

Figure 3. Schematic of the DAT structure.

DAT is inserted between the backbone and RepGFPN, acting on C3, C4, and C5 feature maps to enhance the model's focus on small and occluded targets. Key details are as follows:

(1) Module composition: Each DAT layer includes two multi-head attention blocks and one feed-

forward network (FFN) block. The number of attention heads is configured as 8 (C3), 16 (C4), and 32 (C5) to ensure 32 dimensions per head.

(2) Operator-level architecture: Deformable attention uses a 3 × 3 deformable convolution with a sampling rate of 4 and a deformable group of 8 for flexible irregular region sampling. The FFN block consists of two 1 × 1 convolutions with an expansion ratio of 4 for efficient feature transformation.

(3) Channel dimension configuration: Input and output channel dimensions remain consistent (C3:256 → 256, C4:512 → 512, C5:1024 → 1024) to ensure smooth feature transmission and seamless integration with subsequent RepGFPN.

2.2.3. Loss function improvement

The loss function of bounding box regression greatly impacts target recognition, and a high-quality loss function can greatly enhance the overall performance of the model. Existing loss functions of bounding box regression (e.g., L_{DIoU}, L_{DIoU}, L_{CIoU}, and L_{EIoU}) are predicated with the idea that the aspect ratio of the actual labeled bounding box and that of the predicted bounding box are equal. These loss functions are meaningless if the true labeled aspect ratio differs from the expected one. The loss function of bounding box regression L_{MPDIoU} is defined by minimizing the upper-left and lower-right point distances between the actual labeled bounding box and the predicted bounding box [20]. This definition method not only integrates the factors involved in existing loss functions but also simplifies the computation process.

The schematic of MPDIoU is shown in Figure 4. In the figure, A is the candidate detection frame, B is the real detection frame, and w and h are the width and height of the input image, respectively. (x_1^A, y_1^A) are the coordinates of the lower-left position of A, (x_2^A, y_2^A) are the coordinates of the upper-right position of A, (x_1^B, y_1^B) are the coordinates of the lower-left position of B, and (x_2^B, y_2^B) are the coordinates of the upper-right position of B. d_1^2 are the squares of the Euclidean distances of the upper-left corners of the frames A and B, and d_2^2 are the squares of the Euclidean distances between the points of the lower-right corners of frames A and B.

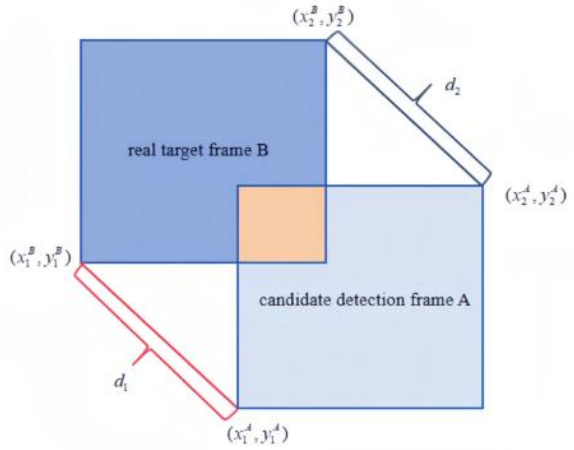

Figure 4. Schematic of MPDIoU.

The MPDIoU loss function is calculated as shown in (6)–(9):

$$d_1^2 = (x_1^B - x_1^A)^2 + (y_1^B - y_1^A)^2, \tag{6}$$

$$d_2^2 = (x_2^B - x_2^A)^2 + (y_2^B - y_2^A)^2, \tag{7}$$

$$MPDIoU = \frac{A \cap B}{A \cup B} - \frac{d_1^2}{w^2+h^2} - \frac{d_2^2}{w^2+h^2}, \tag{8}$$

$$L_{MPDIoU} = 1 - MPDIoU. \tag{9}$$

This loss function can not only assist in selecting the best bounding box for localizing the target but also successfully addresses the issue of detection box distortion caused by overlapping cars.

In conclusion, this improved network based on YOLOv8s is shown in Figure 5, following a workflow of "backbone extraction–fusion enhancement–detection output". Images pass through the backbone network, where CBS, C2f, and SPPF progressively extract and enhance basic and global features. RepGFPN serves as the core fusion layer, employing Up-sampling to align scales, Concat to concatenate multi-scale features, and CSPStage to suppress redundancy, thereby addressing shallow-layer feature loss. DAT is embedded at higher levels to dynamically focus on key features of small or occluded objects. Finally, the detection head predicts vehicle categories and locations based on multi-scale features, adapting to complex nighttime scenarios.

In the model's detection head, the complete intersection over union (CIoU) loss is replaced with the minimum point distance intersection over union (MPDIoU) loss to optimize bounding box regression.

(1) Parameter configuration: The MPDIoU loss adopts the following parameters: distance weight $\alpha = 0.5$, aspect ratio weight $\beta = 0.3$, and missed detection penalty coefficient = 1.2.

(2) Application scope: It is uniformly applied to the three detection layers corresponding to feature map scales 80 × 80, 40 × 40, and 20 × 20, ensuring consistent regression performance across multiscale targets.

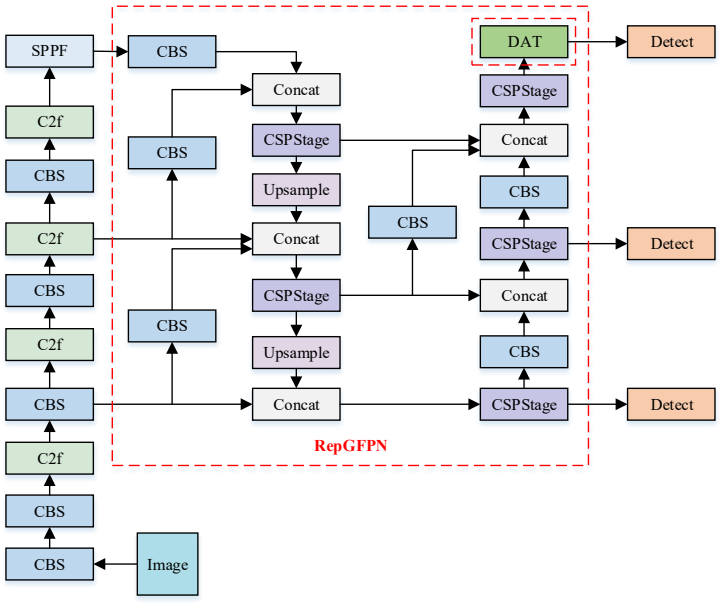

Figure 5. Diagram of the improved YOLOv8s network structure.

3. Experiment and result analysis

3.1. Construction of the experimental dataset

To validate the enhanced algorithm's efficacy for nighttime vehicle detection, a specialized dataset is constructed by integrating filtered public data (BDD100K) and a self-collected domestic dataset.

3.1.1. Filtering of the BDD100K dataset

The BDD100K dataset [21], a large-scale video-based dataset developed by the Berkeley Deep Learning Lab, includes 10 target categories (38% vehicles, 45% pedestrians) with diverse weather and lighting conditions (partial samples in Figure 6).

Figure 6. Partial sample of the BDD100k dataset.

Extraction protocol: Given its video origin, frames are extracted at 10-frame intervals to avoid temporal redundancy. Only frames annotated with "night" scene metadata and "clear/cloudy" weather are selected to ensure consistent lighting backgrounds.

Filtering and annotation standardization: Blurry frames and those without vehicle targets are excluded. Non-vehicle classes are collapsed into a single "Vehicle" category to focus on the core detection task. Original annotations are standardized to align with YOLO format requirements.

Result: 12,524 valid nighttime vehicle frames are retained from 100,000 labeled frames.

3.1.2. Self-built dataset labeling

To bring the trained model closer to the real conditions of domestic night roads, this study captures a video of three domestic cities (covering urban roads, highways, and suburban areas) between 8 pm and 6 am. Lighting conditions include streetlamp illumination, vehicle headlight glare, and weak ambient light. Frames are extracted at five-frame intervals, yielding 3443 initial images. Following a screening process to eliminate any instances of automobiles or severe vehicle overlap, 3276 legitimate photos are ultimately kept. The output YOLO format is chosen for direct usage in the ensuing YOLO network training after the photos are labeled. Figure 7 displays the labeling process interface.

The final dataset (15,800 images: 12,524 from BDD100K + 3276 self-collected) is split using stratified sampling at an 8:1:1 ratio, in which the training set consists of 12,640 images (80%), the validation set of 1580 images (10%), and the testing set of 1580 images (10%). Stratified sampling preserves the distribution of vehicle subcategories, lighting conditions, and scenes across splits, ensuring the reliability of model performance assessments.

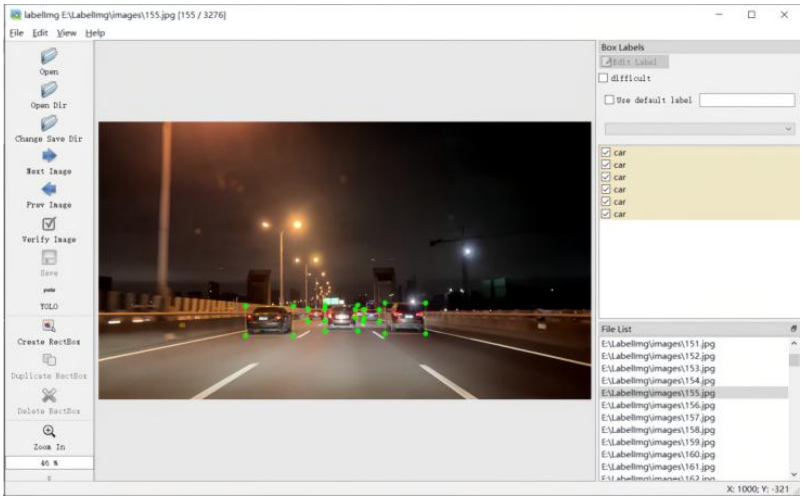

Figure 7. Example of labeling of collected data.

3.2. Experimental environment

In this study, several experiments are conducted on the established hybrid dataset to verify the effectiveness of the improved algorithm. All experimental environments for nighttime vehicle detection are shown in Table 1.

Table 1. Experimental environment.

Hardware devices and dependency libraries	Configuration version
Operating system	Ubuntu 18.04
CPU	Intel Core i5-6400@2.70 GHz
Video card	NVIDIA GeForce RTX 2060 12G
Python	3.8.18
Pytorch	1.11.0
CUDA	10.2
cudnn	8.3

Given the complexity of the nighttime vehicle dataset, which includes low-light conditions, headlight glare, and dynamic interactions between multiple vehicles, as well as the challenge of accurately capturing vehicle contours and details in nighttime vehicle detection, the training parameter settings and optimization process are as follows: The input image size is set to 640 × 640, which retains key features of nighttime vehicles while being compatible with the model's feature extraction network, thereby avoiding computational resource waste caused by overly large sizes. The core hyperparameters and training configurations adopted in our experiments are specified as follows: For the optimizer, we employ the AdamW optimizer with an initial learning rate (LR) of 1e-2, a weight decay coefficient of 5e-4, and beta parameters set to (0.9, 0.999). For the learning rate scheduler, a cosine annealing scheduler is used with a maximum training cycle of 300 epochs and a minimum LR of 1e-5. No warm-up phase is applied during the training process. For data augmentation, two categories of augmentation strategies are integrated: (1) geometric transformations, including random flipping, rotation, and scaling, and (2) image enhancement techniques, such as random adjustment of brightness and contrast, addition of Gaussian noise, and adaptive histogram equalization. To ensure the reproducibility of experimental results, the random seed is fixed to 42 for all experiments.

3.3. Comparative analysis of experimental results

Experiments on the improved YOLOv8s algorithm are trained on the dataset constructed in this study to quantitatively analyze its effectiveness. Figure 8 shows the number of training rounds and the change in accuracy. A high accuracy is achieved during training up to the 30th epoch. A slight fluctuation is observed over the next 120 epochs. Finally, it tends to stabilize gradually, reaching the highest accuracy. The aforementioned experimental findings show that the enhanced YOLOv8s algorithm operates satisfactorily in this work.

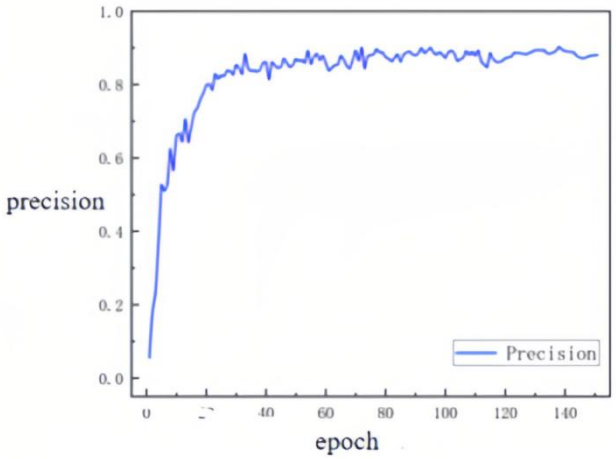

Figure 8. Accuracy curve during training.

Table 2 presents analysis results for YOLOv8s, improved YOLOv8s, and other representative models. The enhanced YOLOv8s network significantly improves the overall detection performance by increasing precision by 6.4%, recall by 4.9%, and mean average precision (mAP) 0.5 by 4.4%, while reducing latency by 6.3 ms, and N_Params by 2.5 million. These improvements stem from RepGFPN's enhanced feature fusion, DAT's focus on key features to boost small object detection capability, and MPDIoU's optimization for regression accuracy. Additionally, tailored datasets and optimized training strategies further contributed to the model's performance gains when compared with the original YOLOv8s network.

To validate the effectiveness of the three proposed improvements to the YOLOv8s algorithm (introducing the RepGFPN module into the feature fusion network, introducing the DAT attention mechanism into the detection network, and replacing the loss function with MPDIoU and their combinations), this study conducted ablation experiments. Ablation experiments systematically remove or modify a specific module within the model to assess the impact and effectiveness of that component on performance. Experiments were conducted on a self-built nighttime vehicle dataset, and the experimental environment and model training details were consistent across all groups to ensure the reliability and comparability of the results. The experimental results are shown in Table 3. The first group represents the detection results of the original YOLOv8s model, serving as the baseline reference; the second, third, and fourth groups represent the results after integrating the RepGFPN module, DAT attention mechanism, and MPDIoU loss function as single modules, respectively. Compared to the original algorithm, the accuracy improved by 3.3%–4.8%, recall improved by 1.9%–3.5%, FPS reduced from 1.4 ms to 4.1 ms, and the N_Params reduced from 0.4 M to 1.7 M. These findings indicate that every single module can effectively improve nighttime vehicle detection performance. The fifth, sixth, and seventh groups represent the results of combining two modules. As shown in the table, compared to integrating a single module, combining two modules yields a slight performance

improvement, reflecting a certain degree of synergy between the modules. The eighth group combines all three modules. The results demonstrate that this approach improves accuracy, recall, and mAP 0.5 by 6.4%, 4.9%, and 4.4% respectively, while reducing FPS by 6.3 ms and significantly decreasing N_Params, thereby substantially enhancing overall model performance. This fully validates the effectiveness of combining the three improved modules and further reinforces the detection advantages of the proposed algorithm in complex nighttime scenarios.

Table 2. Comparison chart of models.

MODEL	P (%)	R (%)	mAP0.5 (%)	FPS (ms)	N_Params (M)
YOLOv7	83.6	72.3	74.4	23.2	13.3
RT-DETR	82.5	75.2	76.8	21.9	14.3
YOLOv8s	84.3	78.6	78.1	22.8	12.1
MITVF [8]	89.4	82.1	81.3	18.2	10.5
Improved YOLOv8s	90.7	83.5	82.5	16.5	9.6

Table 3. Comparison of results from the ablation experiment.

	RepGFPN	DAT	MPDIoU	P (%)	R (%)	mAP0.5 (%)	FPS (ms)	N_Params (M)
1	×	×	×	84.3	78.6	78.1	22.8	12.1
2	√	×	×	87.7	80.5	79.8	21.4	11.7
3	×	√	×	89.1	82.1	81.6	18.7	10.4
4	×	×	√	88.2	81.8	80.9	20.1	11.1
5	√	√	×	89.0	82.4	81.9	18.1	10.2
6	×	√	√	89.5	82.7	82.3	17.2	10.0
7	√	×	√	89.3	82.6	82.1	17.6.	9.8
8	√	√	√	90.7	83.5	82.5	16.5	9.6

3.4. Visualization of experimental results

The detection effect of the optimized YOLOv8s network is visualized and compared with that of the original YOLOv8s network. A comparison of the detection effect is shown in Figure 9, in which the left side shows the pre-optimization effect, and the right side the post-optimization effect.

(a) Before improvement (b) After improvement

Figure 9. Vehicle target detection before and after optimization.

4. Conclusions

The enhanced YOLOv8s-based nocturnal vehicle identification algorithm designed in this study presents the following advantages: The algorithm implements the RepGFPN feature fusion network, which combines re-parameterization and ELAN technology to effectively address the issue of shallow small-scale features being easily lost in the deep network. It introduces the DAT attention mechanism and models the key areas of the feature map with a variable attention model, which improves the recognition ability of the network for small-scale and occluded targets. The proposed algorithm optimizes the regression for the detection frame distortion caused by vehicle overlapping using the MPDIoU loss function. This approach effectively reduces the omission of vehicle detection at night. The experimental evaluation shows that the algorithm maintains the detection speed while increasing the precision by 6.4%, recall by 4.9%, and mAP0.5 by 4.4%, which significantly improves the overall performance.

Although the improved YOLOv8s nighttime vehicle detection algorithm proposed in this study addresses issues such as shallow feature loss, insufficient recognition of occluded targets, and bounding box distortion under low-light conditions through the synergistic optimization of RepGFPN, DAT attention mechanisms, and MPDIoU loss functions, the following limitations remain: At the dataset level, the existing sample repository—which combines curated BDD100K data with independently collected samples—lacks special scenarios such as tunnels and unlit rural roads. It also excludes extreme weather conditions like heavy rain and dense fog, while data for specific vehicle types (motorcycles and large trucks) remains insufficient, limiting generalization capabilities. For extreme scenario adaptation, feature extraction capability is insufficient in extremely low-light conditions. Object discrimination and bounding box localization accuracy require improvement in scenarios where vehicles overlap by over 50% or are occluded by non-vehicle obstacles. For deployment applications, the 9.6 million parameters still struggle to meet the lightweight requirements of low-power in-vehicle edge devices. The current 16.5 ms latency is based on high-performance GPU testing, and edge-side real-time performance requires practical validation. The MPDIoU loss function is sensitive to labeling errors in nighttime blurry scenes and prone to overfitting inaccurate labels, which affects detection stability.

Future research will focus on targeted optimizations: First, expand real-world datasets across multiple regions, scenarios, and weather conditions, and combine digital twin-generated synthetic data with semi- or weakly-supervised learning to reduce reliance on high-precision annotations. Second, enhance the algorithmic framework by introducing adaptive multi-scale augmentation, visual-infrared millimeter-wave radar multimodal fusion, and instance segmentation-assisted localization to strengthen feature extraction and target discrimination in extreme scenarios. Third, adopt compression strategies like quantization, pruning, and knowledge distillation, and optimize networks and operators for vehicle edge computing constraints to balance accuracy and lightweighting. Fourth, refine loss functions by incorporating noise-robustness terms and developing automated annotation correction systems. This enhances model tolerance to labeling errors, accelerating algorithm deployment in autonomous and advanced driver-assistance systems.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported in part by Guangzhou City University of Technology Youth Research Fund Project 52-K0224026.

Conflict of interest

The authors declare no conflict of interest.

References

1. R. Litoriya, K. Bandhu, S. Gupta, I. Rajawat, H. Jagwani, C. Yadav, Implementing visual assistant using YOLO and SSD for visually-impaired persons, *Journal of Automation, Mobile Robotics and Intelligent Systems*, **17** (2023), 79–87. https://doi.org/10.14313/jamris/4-2023/33
2. I. Ogunrinde, S. Bernadin, Improved deepSORT-based object tracking in foggy weather for AVs using sematic labels and fused appearance feature network, *Sensors*, **24** (2024), 4692. https://doi.org/10.3390/s24144692
3. A. Sitepu, C. Liu, Optimized visual recognition through a deep convolutional neural network with hierarchical modular organization, *IEEE Access*, **12** (2024), 95517–95528. https://doi.org/10.1109/ACCESS.2024.3426350
4. M. Soltani-Gol, A. Asgharzadeh-Bonab, H. Soltanian-Zadeh, J. Mazlum, RDCU-Net: a multi-scale residual dilated convolution U-Net with spatial pyramid pooling for brain tumor segmentation, *AUT J. Elec. Eng.*, **56** (2024), 203–212. https://doi.org/10.22060/eej.2023.22395.5538
5. M. Shyamala Devi, S. Alex David, S. Vinoth Kumar, M. Sandeep Prasan Kumar, S. Rohith, Fast-RCNN coupled four-dense layered deep fully connected neural network-based insulator chain defect defection, In: *Proceedings of international conference on recent trends in computing*, Singapore: Springer, 2024, 513–524. https://doi.org/10.1007/978-981-97-1724-8_44
6. J. Wang, Y. Chen, X. Ji, Z. Dong, M. Gao, Z. He, SpikeTOD: a biologically interpretable spike-driven object detection in challenging traffic scenarios, *IEEE Trans. Intell. Transp.*, **25** (2024), 21297–21314. https://doi.org/10.1109/TITS.2024.3468038
7. J. Wang, Y. Chen, X. Ji, Z. Dong, M. Gao, C. Lai, Vehicle-mounted adaptive traffic sign detector for small-sized signs in multiple working conditions, *IEEE Trans. Intell. Transp.*, **25** (2023), 710–724. https://doi.org/10.1109/TITS.2023.3309644
8. J. Wang, Y. Chen, X. Ji, Z. Dong, M. Gao, C. Lai, Metaverse meets intelligent transportation system: an efficient and instructional visual perception framework, *IEEE Trans. Intell. Transp.*, **25** (2024), 14986–15001. https://doi.org/10.1109/TITS.2024.3398586
9. L. Encío, D. Fuertes, C. Del-Blanco, I. Aguilar, C. Pérez-Benito, A. Jevtić, et al., Enhanced nighttime vehicle detection for on-board processing, *IEEE Access*, **13** (2025), 44817–44835. https://doi.org/10.1109/ACCESS.2025.3548837
10. F. Zhang, Nighttime vehicle detection algorithm based on improved YOLOv7, *IEEE Access*, **13** (2025), 126043–126051. https://doi.org/10.1109/ACCESS.2025.3587717
11. Z. Aldoski, C. Koren, Traffic sign detection and quality assessment using YOLOv8 in daytime and nighttime conditions, *Sensors*, **25** (2025), 1027. https://doi.org/10.3390/s25041027
12. Y. Jiang, Z. Tan, J. Wang, X. Sun, M. Lin, H. Li, Giraffedet: a heavy-neck paradigm for object detection, arxiv: 2202.04256. https://doi.org/10.48550/arXiv.2202.04256

13. A. Shaikh, S. Amin, M. Zeb, A. Sulaiman, M. Al Reshan, H. Alshahrani, Enhanced brain tumor detection and segmentation using densely connected convolutional networks with stacking ensemble learning, *Comput. Biol. Med.*, **186** (2025), 109703. https://doi.org/10.1016/j.compbiomed.2025.109703
14. X. Xu, Y. Jiang, W. Chen, Y. Huang, Y. Zhang, X. Sun, Damo-yolo: a report on real-time object detection design, arxiv: 2211.15444. https://doi.org/10.48550/arXiv.2211.15444
15. C. Wang, M. Jia, M. Li, C. Bao, W. Jin, Attention is all you need for blind room volume estimation, *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, 1341–1345. https://doi.org/10.1109/ICASSP48485.2024.10447723
16. L. Yin, L. Wang, S. Lu, R. Wang, Y. Yang, B. Yang, et al., Convolution-transformer for image feature extraction, *Comput. Model. Eng. Sci.*, **141** (2024), 87–106. https://doi.org/10.32604/cmes.2024.051083
17. W. Wang, E. Xie, X. Li, D. Fan, K. Song, D. Liang, et al., Pyramid vision transformer: a versatile backbone for dense prediction without convolutions, *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, 548–558. https://doi.org/10.1109/ICCV48922.2021.00061
18. Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, et al., Swin transformer: Hierarchical vision transformer using shifted windows, *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, 9992–10002. https://doi.org/10.1109/ICCV48922.2021.00986
19. Z. Xia, X. Pan, S. Song, L. Li, G. Huang, Vision transformer with deformable attention, *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, 4784–4793. https://doi.org/10.1109/CVPR52688.2022.00475
20. S. Ma, Y. Xu, Mpdiou: a loss for efficient and accurate bounding box regression, arxiv: 2307.07662. https://doi.org/10.48550/arXiv.2307.07662
21. F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, et al., Bdd100k: a diverse driving dataset for heterogeneous multitask learning, *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, 2633–2642. https://doi.org/10.1109/CVPR42600.2020.00271

AIMS Press

©2026 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0)