ISSN 2616-5775 Vol. 9, Issue 2: 11-17, DOI: 10.25236/AJCIS.2026.090202

GCD-YOLO: Enhanced Vehicle Detection in Low-Light Conditions via Edge Information Transfer and Dynamic Head

Zhen Liu[1,a,*]

[1]*School of Computer Science and Engineering, Anhui University of Science and Technology, Huainan, China*
[a]*415604223@qq.com*
[*]*Corresponding author*

Abstract: *To improve the accuracy of vehicle detection for automotive autonomous driving systems under low-illumination, low-contrast, and dynamic lighting interference conditions at night, this paper introduces a lightweight enhanced model named GCD-YOLO based on the YOLOv12 architecture, focusing on strengthening multi-scale feature extraction and fusion mechanisms. First, to enhance the model's sensitivity to contours and edges and enable learning of richer image feature representations, a Global Edge Information Transfer (GEIT) network is proposed to refine the backbone network, thereby improving detection accuracy. Second, the bottleneck module (A2C2F) is redesigned by incorporating a Content Aware Mixer (CAMixer) module, which achieves efficient and high-quality image super-resolution reconstruction while facilitating better feature fusion. Finally, the original detection head is replaced with a Dynamic Detection Head (DyHead) to mitigate the negative impact of redundant information generated during feature fusion and to increase inference speed. Experimental results demonstrate that GCD-YOLO achieves an accuracy of 94.7% on the BBD100k dataset, representing a 4.6% improvement over YOLOv12, with a detection frame rate of 59.8 frames per second, thus balancing detection accuracy and speed.*

Keywords: *Vehicle Detection, Low-light Environments, Feature Fusion, Yolov12*

1. Introduction

Vehicle object detection is a significant research direction in the field of computer vision. With the increasing application of autonomous driving in vehicles, the requirements for object detection have also risen accordingly. Although vehicle object detection has advanced rapidly in recent years, with continual improvements in recognition rate and speed, its accuracy and stability are still severely compromised in extreme nighttime scenarios. This is primarily due to the poor quality and uneven illumination of the images captured under such conditions [1].

Traditional nighttime vehicle detection techniques often rely on extracting headlights as key features. For instance, Zhang et al. [2] proposed a dual-camera fusion-based headlight detection method. By constructing a bidirectional energy function through geometric distance modeling, the method sequentially achieves headlight pairing, missing information reconstruction, and contour optimization, thereby enabling robust vehicle detection and tracking at night. Lei [3] introduced an adaptive region-of-interest extraction method based on vehicle shadow features, a multi-feature fusion decision mechanism, and a multi-feature matching algorithm for nighttime headlights, enhancing the accuracy and robustness of vehicle detection under complex weather and nighttime conditions. Song [4] incorporated a Squeeze-and-Excitation module into YOLOv5s to enhance feature extraction capability, and integrated a dual-pyramidal feature fusion structure along with replacing the loss function with CIoU Loss, thereby improving both the real-time performance and detection accuracy of the algorithm. Gao et al. [5] proposed a bio-inspired image enhancement method combined with the Faster-RCNN model. By leveraging adaptive feedback mechanisms from retinal horizontal cells and the center-surround antagonism of bipolar cells, they enhanced the contrast and brightness of nighttime images, highlighting headlight contour features and achieving high-precision detection of vehicle headlights in low-light environments.

In summary, while existing improvements have enhanced the overall detection performance of various models, they still face significant challenges in practical complex scenarios. Specifically, in

ISSN 2616-5775 Vol. 9, Issue 2: 11-17, DOI: 10.25236/AJCIS.2026.090202

environments with extreme congestion, severe occlusion, and on nighttime roads (such as urban streets and highways) disturbed by strong road lighting and oncoming headlights, models commonly suffer from declining detection accuracy and increased rates of false positives and missed detections. To address these issues, this paper proposes a lightweight GCD-YOLO model based on the YOLOv12 network, with a focus on comprehensively enhancing the extraction and fusion of key features in low-light scenarios.

2. GCD-YOLO Algorithm

This paper constructs a Global Edge Information-Aware Nighttime Vehicle Detection Model based on YOLOv12. First, to enhance the attention to object edge information at each scale, the features extracted from different layers of the backbone network are fused through a Global Edge Information Transfer Network. Next, to mitigate the decline in detection accuracy caused by insufficient contextual information, a Content-Aware Mixer is designed to accurately identify complex and simple regions within the image and adaptively blend convolution and self-attention based on content complexity. During prediction, a Dynamic Detection Head is employed to reduce the limitation on capturing information from distant positions while enabling convolutional kernels to adopt arbitrary parameter configurations and sampling patterns. Figure 1 illustrates the architecture of the proposed GCD-YOLO network, which consists of four components: Input, Backbone, Neck, and Head.

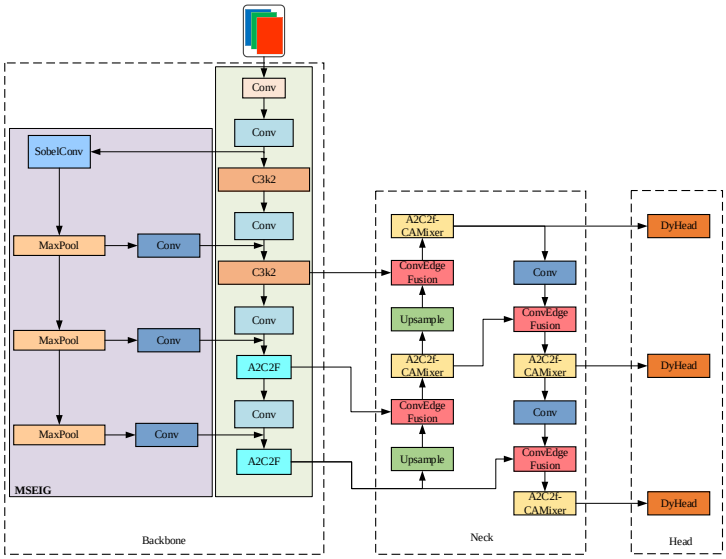

Figure 1: GCD-YOLO network structure.

3. Method

3.1. Global Edge Information Transfer (GEIT)

It is well known that the localization of object bounding boxes heavily relies on edge information. However, conventional object detection networks, such as the YOLO series, lack dedicated components to enhance the model's focus on edge details. To address this limitation, it is necessary to develop a mechanism that effectively integrates edge information into features extracted at multiple scales. Therefore, we propose a structure named Global Edge Information Transfer (GEIT), which facilitates the propagation of edge information extracted from shallow features across the entire network and enables its fusion with features at different scales.

Multi-scale Edge Info Generator (MSEIG). Since the original image contains substantial background information, directly extracting edge information from it and propagating it throughout the entire backbone network would introduce noise into the learning process. Moreover, shallow convolutional layers help filter out unnecessary background details. Therefore, we design a module named the Multi-scale Edge Info Generator (MSEIG) at the shallow stage of the network. This module leverages shallow feature layers to generate multi-scale edge information feature maps, which are then injected into and fused with features at various scales across the backbone network.

Regarding the choice of downsampling methods, careful consideration is necessary. Our objective is

ISSN 2616-5775 Vol. 9, Issue 2: 11-17, DOI: 10.25236/AJCIS.2026.090202

to preserve and enhance edge information while performing downsampling. In this context, MaxPooling is more suitable than AvgPooling. MaxPooling retains the strongest features within local regions, thereby better preserving edge details, whereas AvgPooling is more appropriate for scenarios requiring feature smoothing or homogenization, but tends to be less effective in maintaining fine details and edge information. As illustrated in Figure 1, three MaxPool layers are employed here. This is because, to improve scale adaptability within the MSEIG module, we adopt three distinct scales, necessitating three MaxPool operations with a stride of 2. Performing these operations directly on the 8× downsampled basis enhances network efficiency, as it reduces computational complexity compared to the original model's approach of calculating edge features separately at 2×, 4×, and 8× downsampling stages.

Conv Edge Fusion Module. For the pyramid fusion stage, Conv Edge Fusion intelligently integrates edge information with conventional convolutional features, proposing a novel cross-channel feature fusion strategy. First, it employs conv_channel_fusion to perform cross-channel fusion between edge information and ordinary convolutional features, helping the model better integrate features from different sources. Subsequently, conv_3x3_feature_extract is utilized to further extract the fused features, enhancing the model's capability to capture local details. Finally, conv_1x1 adjusts the output feature dimensions.

There are two primary reasons for placing the Conv Edge Fusion module in the pyramid structure rather than in the backbone. First, although the MSEIG module in the backbone optimizes edge information, such information can gradually diminish after passing through multiple convolutional layers. Therefore, incorporating this module in the pyramid structure further reinforces edge-aware features. Second, it enhances the feature fusion capability of the pyramid structure. Analysis of traditional feature fusion methods reveals that they typically reduce feature dimensions, perform convolution, align features to a uniform scale, and finally concatenate them. While this approach reduces computational load to some extent, it relies solely on standard convolution. In contrast, Conv Edge Fusion adopts an inverse design philosophy: it first expands the number of input channels via pointwise convolution [6], then performs feature extraction using depthwise separable convolution [7], and finally compresses the channels back to the original number through another pointwise convolution. This design allows the intermediate stage to retain a larger number of channels while maintaining linear properties, reducing information loss and facilitating gradient propagation. The structure of Conv Edge Fusion is illustrated in Figure 2

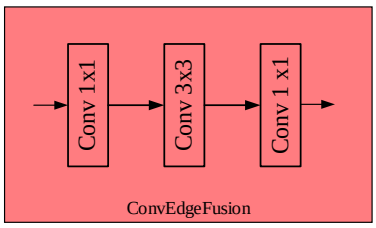

Figure 2: ConvEdgeFusion Structure.

3.2. A2C2F-CAMixer

The YOLOv12 network extensively employs A2C2F modules, whose primary role is to learn residual features. The overall performance of the network heavily relies on the feature-learning capability of these A2C2F modules. However, the A2C2F module is designed with a stronger emphasis on preserving spatial information, making it suitable for tasks involving small objects or those requiring high spatial detail. Yet, when dealing with large objects or complex backgrounds, the A2C2F module may fail to effectively capture contextual information. For instance, in object detection tasks where multiple objects coexist within a complex background, insufficient contextual modeling in the A2C2F module can lead to degraded detection accuracy.

To address this limitation, we propose a novel module named A2C2F-CAMixer. By dynamically allocating convolution and self-attention operations [8], it achieves efficient and high-quality image super-resolution reconstruction. The A2C2F-CAMixer employs a learnable predictor to dynamically assign attention: convolution is allocated for regions with simple contexts, while additional deformable window attention is provided for areas with sparse textures. This mechanism avoids uniform processing of all regions, thereby significantly reducing computational complexity. The structure of the A2C2F-CAMixer is illustrated in Figure 3.

ISSN 2616-5775 Vol. 9, Issue 2: 11-17, DOI: 10.25236/AJCIS.2026.090202

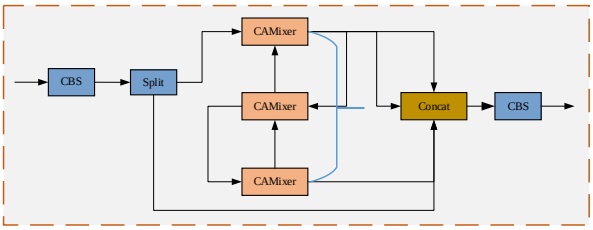

Figure 3: A2C2f-CAMixer Structure.

CAMixer. CAMixer dynamically selects appropriate neural operators based on the complexity of the input image content. For regions with simple context, convolution is applied for processing; for areas with complex textures, deformable window attention is introduced to enhance feature extraction. Simultaneously, CAMixer incorporates a learnable predictor that generates multiple forms of guidance information, including offsets for window deformation, masks for window classification, and convolution attention. This guidance information enables the model to more accurately distinguish between complex and simple regions in the image, thereby achieving more efficient allocation of computational resources. By controlling the proportion of self-attention (γ), CAMixer allows for flexible trade-offs between convolution and self-attention. When $\gamma = 1$, CAMixer relies entirely on self-attention; when $\gamma = 0$, it uses only convolution; and when $0 < \gamma < 1$, CAMixer adaptively blends convolution and self-attention according to content complexity.

3.3. Dynamic Detection Head (DyHead)

DyHead is a unified detection head, whose most prominent feature is the integrated approach to addressing three key challenges: scale awareness, spatial awareness, and task awareness. Although the original detection head of YOLOv12 has been updated compared to its predecessors, it still exhibits certain limitations, such as reduced detection performance in dense object scenes and under occlusion conditions. When images contain a large number of densely arranged objects, detection inaccuracies or missed targets may occur. The detection head of YOLOv12 consists of two branches, each extracting information through two 3×3 convolutions, and finally computing the Bbox loss and Cls loss separately. In the source code, an operation iterating through all channels follows the three convolutional layers, which significantly increases computational complexity.

To address these issues, this paper adopts the Dynamic Head (DyHead) structure [9]. DyHead implements a mechanism that identifies more important channels and better understands the relationships between various positions within an image, thereby effectively integrating global pixel information to enhance the performance of single-image super-resolution reconstruction. As shown in Figure 4, unlike the original fixed dual-branch computation structure, DyHead treats the input as a tensor with three dimensions: level × space × channel, and incorporates an attention mechanism on this tensor. Notably, the three types of attention are integrated into a single detection head, enabling more comprehensive feature fusion without increasing computational cost.

At the final stage of feature fusion, DyHead reduces the constraints on acquiring information from distant locations and allows convolutional kernels to adopt arbitrary parameter configurations and sampling patterns. This provides an adaptive solution capable of effectively handling objects with diverse shapes and varying sizes.

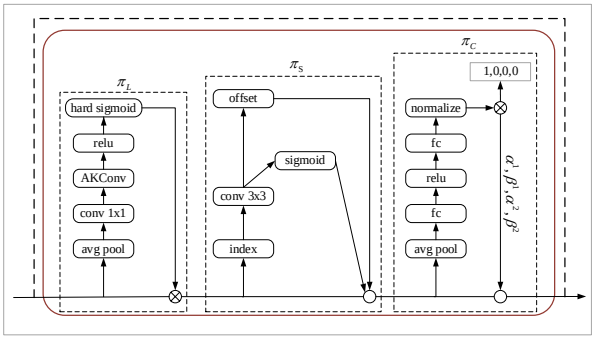

Figure 4: DyHead Structure.

ISSN 2616-5775 Vol. 9, Issue 2: 11-17, DOI: 10.25236/AJCIS.2026.090202

4. Experimental Analysis

4.1. Dataset and Evaluation Metrics

In deep learning research, the selection of datasets plays a crucial role, as dataset quality profoundly impacts the performance of network models. This study employs the BBD100k dataset [10]. From the original dataset, the vehicle category is selected as the detection target, and samples covering various nighttime scenarios such as urban roads, highways, and residential areas are included, encompassing both moving and stationary vehicle states. The data is split into training and test sets in a 4:1 ratio, consisting of 4,000 training images and 1,000 test images, while ensuring a balanced distribution of scenario types across both sets. Sample images from the constructed low-light vehicle dataset are illustrated in Figure 5.

Figure 5: Example of BBD100k dataset.

In this experiment, the following commonly used evaluation metrics for object detection performance were adopted: Precision, Recall, Average Precision (AP), mean Average Precision (mAP), and Frames Per Second (FPS)[11,12]. Precision indicates the proportion of correctly predicted positive samples among all samples predicted as positive by the model, and its calculation formula is provided in Equation (1). Recall measures the proportion of correctly identified true targets relative to all actual targets, as expressed in Equation (2). The mAP is computed as the average of AP values across all categories, defined by Equation (4).

$$Percision = \frac{TP}{TP+FP} \tag{1}$$

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

$$AP = \frac{TP+TN}{TP+TN+FP} \tag{3}$$

$$mAP = \frac{\sum_{i=1}^{k} AP}{k} \tag{4}$$

$$FPS = \frac{Frames}{Time} \tag{5}$$

4.2. Ablation Studies

Ablation studies are a commonly used experimental approach in deep learning, aimed at understanding the impact of individual components on model performance. Through ablation experiments, it is possible to identify which parts of the model are effective, which are redundant, and to assess the relative importance of each component. To ensure comparability, all models in this study were trained and evaluated under identical conditions, including the same dataset, hyperparameters, and training-testing procedures. This controlled setup allows for a clear distinction between the performance differences between the proposed GCD-YOLO model and the baseline YOLOv12 model.

This paper introduces improvements to the YOLOv12 algorithm by integrating the GEIT structure, proposing the A2C2f-CAMixer module, and adopting DyHead (denoted as G, C, and D, respectively, in Table 1. During training, GPU memory consumption with a batch size of 1 is recorded as GPU Memory.

Table 1: Ablation experiment.

Models	*Params/MB*	*GPU Memory/G*	*GFLOPS*	*mAP@0.5/%*	*mAP@0.95/%*	*FPS(f s-1)*
YOLOv12	2.56	3.1	6.3	90.1	56.3	60.0
+G	2.77	4.3	7.3	92.8	57.6	51.8
+G+C	2.66	3.9	7.8	93.6	58.3	55.5
+G+C+D	2.67	3.3	6.9	94.7	60.0	59.8

The ablation results are presented in Table 1. They indicate that each proposed modification to

ISSN 2616-5775 Vol. 9, Issue 2: 11-17, DOI: 10.25236/AJCIS.2026.090202

YOLOv12 influences both accuracy and mAP@0.5. Using YOLOv12 as the baseline model (Group 1), it achieves 2.56 MB parameters, 3.1 GB GPU memory, 6.3 GFLOPS computational cost, with mAP@0.5 of 90.1%, mAP@0.95 of 56.3%, and an FPS of 60.0. When integrating GEIT into YOLOv12 (Group 2), the model parameters increase to 2.77 MB, GPU memory rises to 4.3 GB, computational cost increases to 7.3 GFLOPS, and mAP@0.5 and mAP@0.95 improve to 92.8% and 57.6%, respectively. While this module enhances accuracy, it also demands higher computational resources, resulting in a reduced FPS of 51.8. Subsequently, incorporating the content-aware C2f-CAMixer module into Group 2 (Group 3) slightly reduces model parameters to 2.66 MB, GPU memory to 3.9 GB, and computational cost to 7.8 GFLOPS, while further improving mAP@0.5 and mAP@0.95 to 93.6% and 58.3%, respectively. The FPS also increases to 55.5, indicating that this structure optimizes both accuracy and computational efficiency. Finally, replacing the original YOLOv12 detection head with DyHead (Group 4) keeps the parameter count nearly unchanged, slightly reduces GPU memory, lowers computational cost to 6.9 GFLOPS, and enhances mAP@0.5 and mAP@0.95 to 94.7% and 60.0%, respectively. This demonstrates its higher adaptability and lightweight design, which further reduces overall processing time. Collectively, the integration of GEIT, A2C2f-CAMixer, and DyHead results in a more efficient model that reduces computational cost while maintaining relatively high performance.

4.3. Comparative Experiments

To validate the superiority and feasibility of the proposed algorithm, the GEIT-YOLOv8 algorithm was evaluated against several state-of-the-art algorithms on relevant datasets, including YOLOv5n [13], YOLOv8n, YOLOv9c [14], CCEI-YOLOv8 [15], and RT-DERT [16]. The experimental results are presented in Table 2.

Table 2: Performance comparison of different detection algorithms on the BBD100k dataset.

Models	*mAP@0.5/%*	*FPS(f s-1)*
YOLOv5n	84.2	187
YOLOv8n	90.1	202
YOLOv9c	86.4	194
CCEI-YOLOv8	91.7	56.4
YOLOv12	90.1	60.0
RTDETR	93.8	57.0
GCD-YOLO	94.7	59.8

According to Table 2, the GCD-YOLO algorithm demonstrates outstanding detection accuracy. On the BBD100k dataset, it achieves an mAP@0.5 of 94.7%, showing a clear advantage over the YOLOv5n and YOLOv8n algorithms. In terms of real-time performance, GCD-YOLO attains a frame rate of 59.8 FPS on the BBD100k dataset, which is second only to other YOLO-series algorithms and significantly outperforms the remaining compared methods.

In this complex scenario, the YOLOv12 algorithm exhibits pronounced false detection issues. In contrast, the GCD-YOLOv8 algorithm, leveraging its robust feature extraction and fusion capabilities, not only effectively mitigates interference from sparsely distributed targets but also accurately detects the intended objects, demonstrating its superior performance under challenging conditions.

In summary, compared to the YOLOv12 algorithm, the GCD-YOLOv8 algorithm significantly enhances detection performance and robustness when handling complex and variable scenes.

5. Conclusion

To address the issues of target occlusion and the high rates of missed or false detections for small-scale objects in construction site scenarios, this paper presents an improved version of YOLOv12. On one hand, the proposed approach integrates the GEIT structure to enhance local perception and expand global abstraction capabilities. On the other hand, the A2C2f-CAMixer structure is introduced to optimize the extraction of image edge features, enabling the algorithm to effectively capture contour and edge information across targets ranging from small to large scales. Additionally, the DyHead detection head is adopted to dynamically adjust detection strategies, further improving generalization and adaptability.

In future work, we will focus on enhancing the detection capability for targets with low data representation and deploying the optimized algorithm efficiently on embedded devices to improve resource efficiency and real-time performance. Furthermore, the integration of technologies such as

ISSN 2616-5775 Vol. 9, Issue 2: 11-17, DOI: 10.25236/AJCIS.2026.090202

semantic segmentation and instance segmentation will be explored to strengthen the algorithm's practicality and generalization ability, thereby meeting more diverse application demands.

References

[1] FAN J Q, LI X, HUO T J. Research on cross domain detection in intelligent vehicles based on one-stage algorithm[J]. China Journal of Highway and Transport,2022,35(3):249-262(in Chinese).
[2] ZHANG X , STORY B, RAJAN D. Night time vehicledetection and tracking by fusing vehicle parts from multiplecameras[J]. IEEE Transactions on Intelligent TransportationSystems,2021,23(7):8136-8156.
[3] LEI Z. Vehicle detection method research based on multi-feature fusion under complicated environment[D]. Jiangxi: East China JiaotongUniversity,2013:44(in Chinese).
[4] SONG P, et al. Detection and tracking of ground moving target by uav in complex environment[D]. Shanxi: Xi'an TechnologicalUniversity,2023:5-7(in Chinese).
[5] GAO Y, et al. Research on nighttime vehicle detection system using feature fusion CNN [D]. Nanjing: Nanjing University,2018.
[6] Hua B S, Tran M K, Yeung S K, et al. Pointwise convolutional neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 984-993
[7] Chollet F, et al. Xception: Deep learning with depthwise separable convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1251-1258.
[8] Shaw P, Uszkoreit J, Vaswani A, et al. Self-attention with relative position representations[J]. arxiv preprint arxiv:1803.02155, 2018.
[9] Dai X, Chen Y, et al. Dynamic head: Unifying object detection heads with attentions[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 7373-7382.
[10] YU F, CHEN H F, WANG X, et al. Bdd100k: a diverse driving dataset for heterogeneous multitask learning[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020:2636-2645.
[11] Liu W, Quijano K, Crawford M M. YOLOv5-Tassel: Detecting tassels in RGB UAV imagery with improved YOLOv5 based on transfer learning[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2022, 15: 8085-8094.
[12] Wang C, Sun W, Wu H, et al. A low-altitude remote sensing inspection method on rural living environments based on a modified YOLOv5s-ViT[J]. Remote Sensing, 2022, 14(19): 4784.
[13] Zhang Y, Guo Z, Wu J, et al.Real-time vehicle detection based on improved yolo v5[J]. Sustainability,2022,14(19): 12274.
[14] Wang C, I-Hau Y, Hong Y, et al. Yolov9: Learning what you want to learn using programmable gradient information[J]. European conference on computer vision, 2024.
[15] Zhang Y, Liu H, Jia S, et al. Lightweight detection method for safety helmet and reflective vest in construction environments[J/OL]. Electronic Measurement Technology, 2025:1-13.
[16] Sun G, Wang X, Li Y, et al. Improved helmet detection algorithm for two-wheeled vehicles of RT-DETR[J/OL]. Journal of Electronic Measurement and Instrument, 2025: 1-13