

FECS-DETR: A Real-time Complex Road Multi-object Detection Model Improved Based on RT-DETR

Hong Zhang^{1,2*}, Linlin Hu¹, Le Wang¹, Yixuan Li¹, Xiaoyu He¹

¹Transportation Institute of Inner Mongolia University, Hohhot 010070, China

²Inner Mongolia Engineering Research Center for Intelligent Transportation Equipment, Hohhot 010070, China

Keywords: Complex scenarios; RT-DETR; Multi-object detection; Feature enhancement; Intelligent transportation

Citation: Zhang H., Hu L.L., Wang L., et al.: FECS-DETR: A Real-time Complex Road Multi-object Detection Model Improved Based on RT-DETR. Data Intelligence, 2026, Vol. 8, Art. No.: 20260059. DOI: <https://doi.org/10.3724/2096-7004.di.2026.0059>

ABSTRACT

Multi-scale object detection in complex road scenarios remains an arduous challenge, primarily attributed to environmental interference and background heterogeneity. To surmount these obstacles, this paper introduces an enhanced real-time Transformer-based detection model, FECS-DETR. In the design of the backbone network, a feature enhancement architecture is constructed, integrating a lightweight FasterBlock architecture with an Exponential Moving Average (EMA) attention mechanism. The lightweight FasterBlock architecture enables efficient extraction of shallow features by optimizing inter-layer parameter sharing, while the EMA attention mechanism establishes long-range context dependencies through dynamic weight allocation. This synergy significantly enhances the model's spatio-temporal perception in intricate scenarios. For the neck network, the Cross Stage Partial Stage (CSPStage) module is introduced. Leveraging dual mechanisms of multi-scale feature decoupling and noise suppression, CSPStage effectively mitigates redundant information while substantially improving the feature continuity of occluded objects. Experimental results demonstrate that on the SODA10M dataset, FECS-DETR series achieves a 2.3%–4.0% relative growth rate in mean average precision (mAP@50) compared to the baseline RT-DETR model. The lightweight version FECS-DETR-18 has an 18% relative growth rate in Giga Floating-Point Operations (GFLOPs) compared with RT-DETR-18, and its mAP@50 improves by 2.5%. This achievement exemplifies an optimal balance between performance and efficiency. Additionally, on the KITTI and BDD100K datasets, the model attains maximum relative growth rates of 0.9% and 2.2% in mAP@50, respectively validating its excellent generalization ability. Visual analysis vividly showcases the model's outstanding detection performance in scenarios involving overlapping objects and low-light conditions. Through three types of interference experiments, namely Gaussian noise, motion blur, and light mutation, the robustness

*Corresponding author: Hong Zhang (E-mail:zh-hong@imu.edu.cn; ORCID: 0000-0003-3696-6328).

of FECS-DETR against complex environmental perturbations is further verified, confirming the model's superiority in practical applications.

1. INTRODUCTION

With the rapid development of the Intelligent Transportation System (ITS), there is an increasing demand for higher accuracy and real-time performance in object detection technology for autonomous driving [1]. Accurately detecting multi-scale road objects in complex road environments has become a critical task in ITS data analysis. Traditional object detection algorithms face numerous challenges when dealing with multi-scale objects, complex backgrounds, variable weather conditions, lighting variations, and object occlusions [2]. Especially in complex scenarios such as urban streets and highways, improving the detection accuracy of vehicles, pedestrians, traffic signs, and other objects remains an open and challenging issue [3]. Although deep learning-based object detection models, such as the YOLO [4] series and DETR [5], have achieved remarkable progress in the field of object detection by integrating convolutional neural networks (CNNs) and attention mechanisms, their performance still falls short when handling multi-scale objects in complex traffic scenarios, particularly small objects or occluded objects in intricate backgrounds [6]. In recent years, numerous studies have attempted to enhance model performance by improving network architectures [7], introducing lightweight modules [8], and optimizing attention mechanisms. RT-DETR (Real-Time Detection Transformer), as an emerging object detection framework that combines the Transformer architecture with CNNs, has shown great potential in real-time detection [9]. However, previous research has indicated that the backbone network of the RT-DETR model suffers from feature redundancy across channels. This issue not only reduces the model's ability to distinguish between different objects but also leads to insufficient feature extraction and information loss, making it difficult to meet the high-precision requirements of intelligent transportation systems [10]. To address these challenges, this paper proposes an enhanced complex road object detection model, FECS-DETR, based on RT-DETR. The model features a temporal-aware feature enhancement structure that synergistically integrates the noise suppression capability of the Exponential Moving Average (EMA) [11] attention mechanism and the multi-scale feature fusion of the CSPStage module [12], effectively tackling complex challenges such as occlusions in crowded scenes and low-light conditions in nighttime surveillance. The EMA attention mechanism continuously focuses on key feature regions, enhancing the model's perception of important information in complex scenarios. Meanwhile, FECS-DETR employs asymmetric channel processing and feature reuse strategies to reduce computational costs while dynamically compensating for information loss through EMA attention, achieving a balance between lightweight design and high accuracy. Experimental results on multiple public datasets demonstrate that FECS-DETR outperforms state-of-the-art YOLO series and DETR models, verifying its efficiency and robustness in complex road scene detection. Additionally, the model exhibits excellent scalability and can be adapted to different versions of RT-DETR to meet diverse application scenarios, as illustrated in Figure 1.

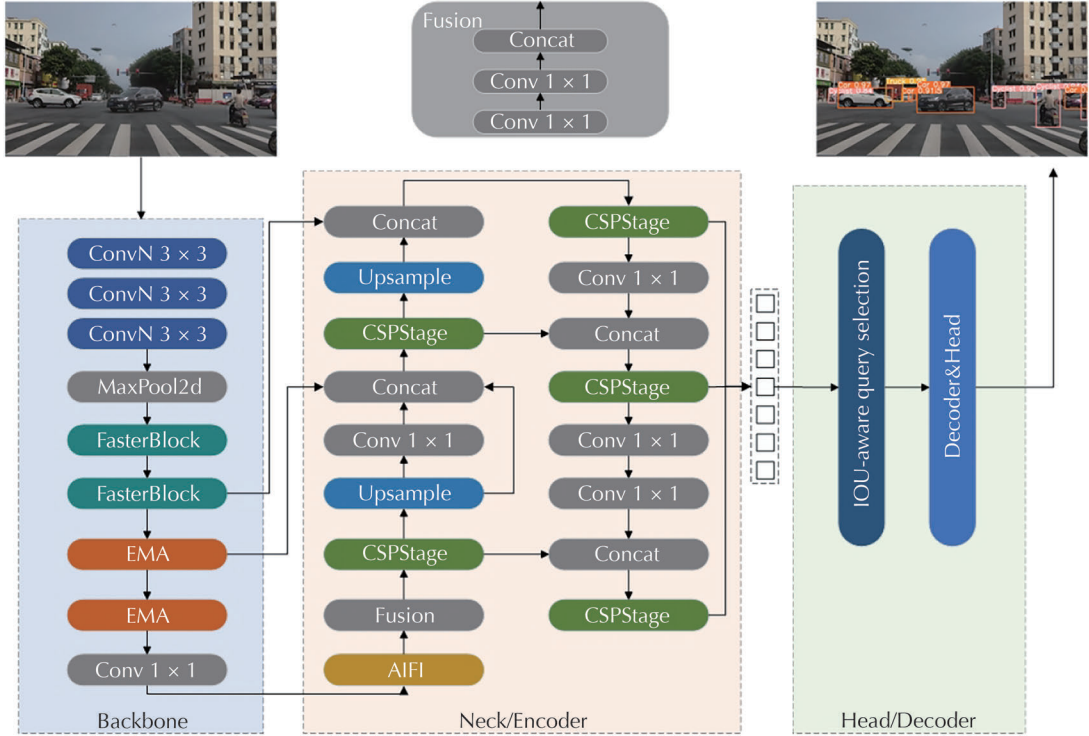


Figure 1. Overview Structure of FECS-DETR.

The main contributions of this paper are summarized as follows:

- (1) **Backbone Network Optimization:** A lightweight FasterBlock [13] module is introduced, which leverages efficient cross-layer connections and dynamic channel adjustment mechanisms. This design significantly enhances feature extraction capabilities while reducing redundant computations, thereby improving the detection accuracy for small and occluded objects.
- (2) **Attention Mechanism Enhancement:** The EMA attention [11] mechanism is integrated into the backbone network to improve classification confidence in complex scenarios, such as occluded or densely distributed objects, through long-range dependency modeling. This mechanism not only enhances the model's robustness to lighting changes but also stabilizes attention weights during training via a moving average strategy, accelerating convergence and reducing false positives and false negatives.
- (3) **Neck Network Innovation:** The CSPStage [12] lightweight feature processing module is adapted and integrated into the RT-DETR neck. By dynamic channel splitting with optimized ratios for $P3/P4/P5$ feature layers and recombination, it reduces the parameters in the feature fusion stage. The divide-and-conquer fusion strategy preserves shallow texture information and deep semantic features, thereby improving the detection performance for small objects and enhancing the feature continuity of occluded objects.

(4) Experimental Validation: Comparative and ablation experiments, along with visualization analysis, are conducted on multiple datasets, including SODA10M [14], KITTI [15] and BDD100K [16]. These experiments comprehensively validate the effectiveness and robustness of FECS-DETR.

The remainder of this paper is organized as follows. Section 2 reviews the related work. Section 3 presents the detailed design and implementation of FECS-DETR. Section 4 reports the experimental results and analyzes the model performance. Finally, Section 5 concludes the paper and discusses its limitations.

2. RELATED WORK

Road perception serves as the cornerstone of autonomous driving systems and traffic flow analysis frameworks, with its core task being the precise extraction and semantic parsing of key elements (e.g., lane markings, traffic signs, dynamic obstacles) from multi-modal sensor data [17]. Early approaches relied on manually designed image processing for infrastructure and object recognition but suffered from poor environmental adaptability [18]. The advent of deep learning has triggered a paradigm shift, enabling data-driven feature learning with enhanced robustness [19], spawning two primary categories of object detection frameworks: two-stage and one-stage architectures.

Two-stage detectors, pioneered by R-CNN [20], employ region proposal algorithms (e.g., selective search [21]) combined with region-based convolutional feature extraction and cascaded classification-regression operations. Subsequent improvements, such as ROI alignment in Fast R-CNN [22] and region proposal networks (RPNs) in Faster R-CNN [23], have significantly enhanced computational efficiency through joint optimization of proposal generation and feature learning. Mask R-CNN [24] further extends this paradigm to instance segmentation tasks. Although these models achieve high accuracy in autonomous driving applications, the cascaded operations inherently introduce substantial computational overhead, hindering their real-time deployment in latency-sensitive vehicular systems [4]. One-stage detectors streamline the process by integrating class prediction and spatial regression into an end-to-end pipeline, eliminating explicit proposal generation. Notable examples include SSD, which utilizes multi-scale feature maps for scale-invariant detection, and the YOLO series, which reformulates detection as gridbased dense prediction [25]. The YOLO architecture's simplicity enables real-time inference while maintaining competitive accuracy, making it well-suited for traffic scenarios demanding immediate responses [26].

More recently, breakthroughs in Vision Transformers (ViTs) have revolutionized detection frameworks. DETR, as the first fully end-to-end Transformer-based detector, eliminates heuristic components (e.g., anchor boxes, non-maximum suppression) through an encoder-decoder architecture and bipartite matching loss [5]. However, its slow convergence and suboptimal small-object performance restrict practical use. Key variants address these limitations: Deformable DETR uses sparse deformable attention to reduce computation and enhance smallobject detection [27]. RT-DETR sets a new benchmark for industrial-level computational efficiency, achieving 65 FPS at 1080p resolution through a hybrid encoder design and adaptive query selection [9].

Notably, the core contribution of MDFD2-DETR [10] lies in proposing the Multi-Domain Feature Decomposition (MDFD) module, which reduces feature redundancy through spatial-channel-frequency domain decoupling. MDFD2-DETR's MDFD module uses a "decomposition-pruning" multi-domain decoupling strategy to reduce feature redundancy and parameters, but suffers from fine-grained feature loss. By contrast, FasterBlock adopts a "selection-aggregation" paradigm: via coordinated partial convolution (PConv) and pointwise convolution (PWConv), it cuts redundant computation, speeds up inference and better retains feature integrity. Recent advances in autonomous driving multi-modal detection underscore the pivotal role of high-quality LiDAR data. For instance, Zhang et al. [28] proposed Sand-CycleGAN, a physics-constrained generative adversarial network that synthesizes the Sand-VisDrone dataset, to address object detection performance degradation under sandstorms (a representative harsh road-related weather), and this dataset enhances the detection robustness of models like YOLOv10n and RT-DETR. The study Denoise Yourself [29] proposes a self-supervised point cloud upsampling framework: using pretrained denoising diffusion models to resolve LiDAR sparsity and noise, it reconstructs geometric details via implicit neural representation without paired annotations. This study not only fills the gap of multi-modal detection literature in the related work but also establishes a technical link between our single-modal optimization (FECS-DETR) and future multi-modal extensions.

Despite these advancements, complex road environments pose critical challenges: significant performance degradation due to object scale variations and dense occlusions, feature distribution shifts caused by day-night lighting changes, and computational redundancy from indiscriminate feature activation in multi-object scenarios [10]. To overcome these limitations, this study presents FECS-DETR, a Transformer based road object detection algorithm. By leveraging temporal-aware feature enhancement, FECS-DETR suppresses redundant computations while preserving the geometric fidelity of occluded and small objects, enabling synergistic optimization of global context modeling and detection accuracy. Experimental results demonstrate that FECS-DETR outperforms current state-of-the-art methods across multiple datasets, achieving higher accuracy with fewer GFLOPs and parameters (Figure 2).

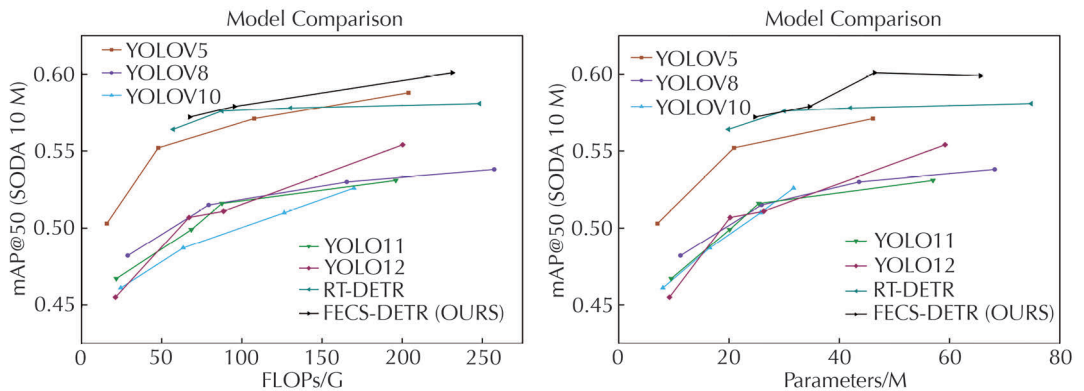


Figure 2. Comparison of results on the SODA10M dataset between the method proposed in this paper and classic realtime road perception methods.

3. THE PROPOSED APPROACH

In road object detection, accurate object feature extraction is pivotal for ensuring the performance of perception algorithms. However, existing detection methods still face significant technical challenges when handling real-time detection of multi-scale objects in complex scenarios. To address these limitations, we propose the FECS-DETR model, an enhanced version of RT-DETR, which constructs an efficient road object detection network through systematic optimizations of the RT-DETR backbone and neck layers. The model incorporates three key innovations: the introduction of the FasterBlock module and Exponential Moving Average (EMA) attention mechanism in the backbone network, and the reconstruction of the neck network using the CSPStage module. These enhancements significantly improve detection accuracy while effectively suppressing noise interference during feature fusion. Each module is detailed as follows.

3.1 The FasterBlock Module

The FasterBlock module[13] is designed based on partial convolution (PConv) and dynamic multi-path feature fusion strategies, enabling lightweight design while enhancing feature representation capabilities. In this study, re-parameterizable FasterBlock modules are integrated into the basic residual blocks at the $P2$ and $P3$ stages of the backbone network, balancing feature extraction efficiency and accuracy for real-time object detection tasks. The FasterBlock module first partitions the input feature map via partial convolution (PConv). To quantify this partitioning, a key hyperparameter, the channel splitting ratio α , is introduced, with α value of 0.5. Based on the experimental settings detailed in Section 4 of this paper, we conducted cross-validation experiments using the FECS-DETR-50 model on the SODA10M dataset to determine the optimal value of α . Testing different candidate values of α during cross-validation revealed: when α is 0.5, the model's mAP reaches 60.1%; when α is smaller than 0.5, overly fine channel partitioning causes fragmentation of feature information, leading to a significant drop in mAP to 55.3%; when α is larger than 0.5, the computational cost of the model increases without a corresponding improvement in mAP. To adaptively balance the contributions of convolutional features and shortcut features (especially in occluded or low-light scenarios), a dynamic residual weight λ , a learnable scalar constrained within a specific range, is introduced. During training, λ automatically modulates feature fusion: for example, in regions where object occlusion exceeds 75%, λ converges to approximately 0.6, prioritizing the preservation of shortcut-based textures; in regions with clear object boundaries, λ drops to around 0.4 to emphasize convolution-derived semantic features. Additionally, the FasterBlock module employs a lightweight Multi-Layer Perceptron (MLP) with an inverted bottleneck structure, where the channel expansion factor r of the 1×1 convolutional layers is 4, consistent with the design of the original FasterNet. As illustrated in Figure 3, the module splits input channels via PConv: one part undergoes convolutions, while the other uses skip connections. Leveraging dynamic residual connections and channel-wise MLP, it enhances small object detection and refines global context. The structure expands channels with 1×1 convolutions, introduces non-linearity via ReLU, then compresses channels back via another 1×1 convolution to optimize feature representation.

$$F_{\text{partial}} = \text{Conv}_{3 \times 3}(X_{C/n_{\text{dn}}}) \oplus X_{C/n_{\text{dn}}} \quad (1)$$

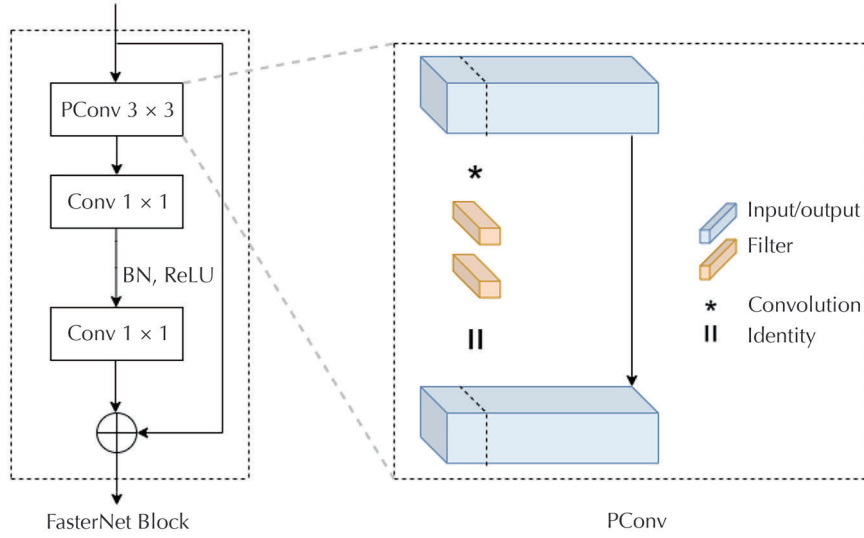


Figure 3. Overall architecture of our FasterNet block. Within each FasterNet block, a PConv layer is followed by two PWConv layers. We put normalization and activation layers only after the middle layer to preserve the feature diversity and achieve lower latency.

In the above equation, X represents the input feature map (tensor). C represents the number of channels of the input feature map X . ndiv represents the parameter of channel segmentation ratio. Moreover, $\oplus$ denotes the channel concatenation operation.

3.2 The EMA Module

The EMA attention mechanism is a lightweight module specifically designed for real-time object detection tasks. It achieves multi-granularity feature enhancement through strategies including grouped channel reconstruction, bidirectional spatial compression, and dynamic dual-path interaction. As illustrated in Figure 4, the mechanism comprises four core stages. Initially, the input feature map $X \in \mathbb{R}^{B \times C \times H \times W}$ is partitioned into G subgroups along the channel dimension. Based on the experimental settings detailed in Section 4 of this paper, we validated through comparative experiments (with group numbers 16, 32, and 64) using our FECS-DETR-50 model on the SODA10M dataset that: when the group number is 16, the mean Average Precision (mAP) reaches 58.9%; when the group number is 32, the mAP attains 60.1%, exhibiting higher precision; and when the group number is 64, the mAP shows no significant improvement. Thus, the group number G of 32 is selected as the optimal choice. After grouping, X is reshaped into $X_g \in \mathbb{R}^{(B \times G) \times (C/G) \times H \times W}$. Subsequently, each subgroup undergoes horizontal and vertical adaptive pooling operations, generating spatial compression descriptors $P_h \in \mathbb{R}^{H \times 1}$ and $P_w \in \mathbb{R}^{1 \times W}$. These descriptors are then merged via a 1×1 convolutional layer to obtain the joint spatial weight matrix W_s , expressed as:

$$W_s = \sigma(\text{Conv}_{1 \times 1}(\text{Concat}(P_h, P_w))) \quad (2)$$

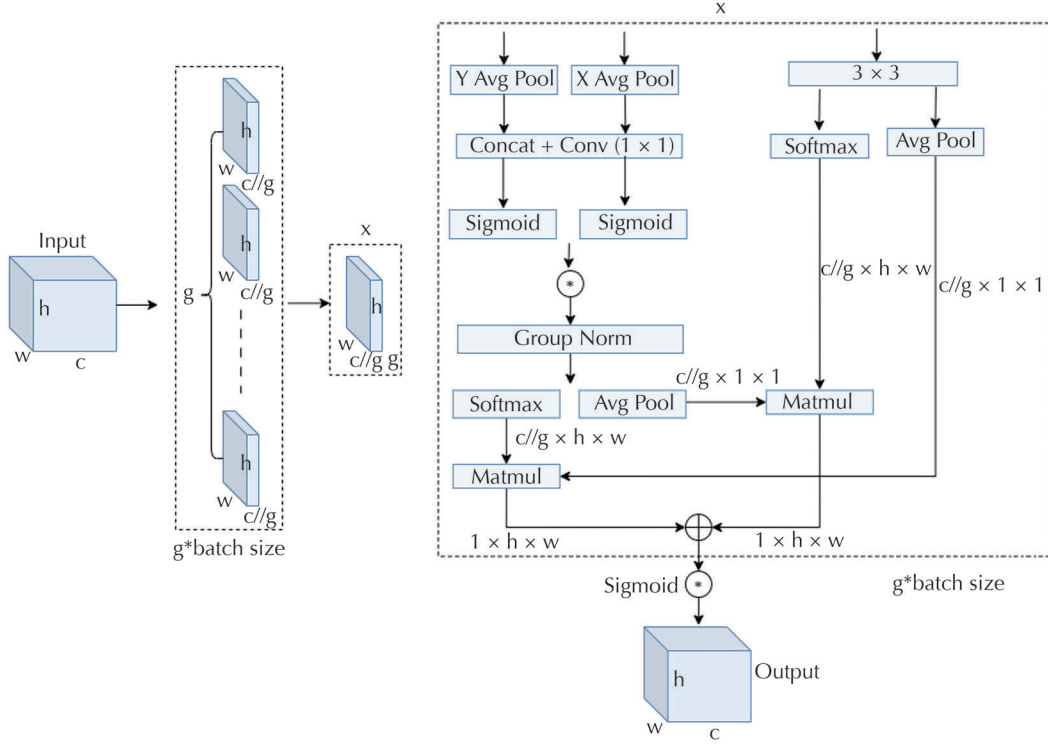


Figure 4. The structure of EMA. Here, g stands for the divided groups, X Avg Pool stands for the 1D horizontal global pooling, $*$ denotes the process of re-weight and Y Avg Pool stands for the 1D vertical global pooling respectively.

In the above equation, σ represents the Sigmoid activation function. This operation effectively encodes two-dimensional spatial dependencies into a compact channel-location joint representation. Upon splitting W_s , horizontal and vertical modulation factors $W_h \in \mathbb{R}^{H \times 1}$ and $W_w \in \mathbb{R}^{1 \times W}$ are obtained. Feature enhancement is achieved through the interaction between the convolutional path and the normalization path. The convolutional path applies 3×3 convolutions to the original grouped features to capture global contextual semantic information F_{com} , while the normalization path extracts local detail features F_{norm} via spatial semantic and Group Normalization. Both feature streams are compressed to the channel dimension using Adaptive Global Pooling (Avg Pool) and then transformed into competitive attention weights A_1 and A_2 through the Softmax function.

For the feature fusion weights between the convolutional path and the normalization path in dynamic dual-path interaction: Attention weights A_1 and A_2 first undergo outer product operations with the transposed feature descriptors F_{norm}^T and F_{com}^T , respectively, followed by summation. Then, the fusion weight matrix W_c is generated via Sigmoid activation as shown in Equation (5). W_c can dynamically balance the contribution ratio of global semantic information from the convolutional path and local detail information

from the normalization path to the final features—placing more emphasis on the fusion of local details in regions dense with small objects, while leaning toward the transmission of global semantics in regions with large objects or background, thereby achieving “on-demand fusion”.

Matrix multiplication between the weight matrices and the spatially expanded features establishes the channel-spatial joint attention, with the computational process defined as follows:

$$F_{\text{norm}} = \text{DWConv}_{3 \times 3} \left(\text{GN} \left(X_g \odot (W_h \otimes W_w) \right) \right) \quad (3)$$

$$F_{\text{conv}} = \text{Conv}_{3 \times 3} (X_g) \quad (4)$$

$$W_c = \sigma \left(A_1 F_{\text{norm}}^T + A_2 F_{\text{conv}}^T \right) \quad (5)$$

In the above equation, $\odot$ denotes element-wise multiplication, $\otimes$ represents the outer product operation. Depthwise separable convolutions are employed, reducing the parameter count to one-ninth of that of standard convolutions and decreasing the computational complexity from $O(H^2W^2)$ to $O(HW)$. Finally, the original features are fused with the attention weights via a Sigmoid gating mechanism, selectively activating 15%–40% of the spatial locations and significantly reducing computational redundancy. By preserving multi-granularity information, this mechanism effectively improves the detection performance for small objects, enabling the RTDETR model to achieve a dual breakthrough in real-time processing and high accuracy.

3.3 The CSPStage Module

The original RT-DETR model suffers from significant computational inefficiencies in its neck network. The traditional Transformer decoder head, relying on self-attention mechanisms, incurs substantial computational overhead. Additionally, direct concatenation of multi-scale features not only introduces information redundancy and reduces fusion efficiency but also leads to the loss of high-level feature details, severely degrading the sensitivity for small object detection. To address these issues, this study introduces the CSPStage (Cross Stage Partial Stage) module into the neck network. Inspired by CSPDenseNet (Cross Stage Partial DenseNet), this module optimizes the computational efficiency of dense connections by applying the CSP (Cross Stage Partial) principle. As illustrated in Figure 5, in the CSPDenseNet architecture, the feature map at each stage is divided into two parts (X_0 and X'_0) along the channel dimension. X_0 is directly transmitted to the final layer of the stage, while X'_0 undergoes processing through a Dense Layer followed by a transition layer. The output of this process is then concatenated with X_0 , and the combined feature map is further processed by another transition layer to generate X_T . This output is subsequently concatenated with X_0 of the next stage and fed into the subsequent transition layer, resulting in the final output X_L . This design preserves the feature reuse advantages of DenseNet while preventing the accumulation of redundant gradient information through gradient flow truncation, achieved via hierarchical feature fusion strategies applied in local transition layers.

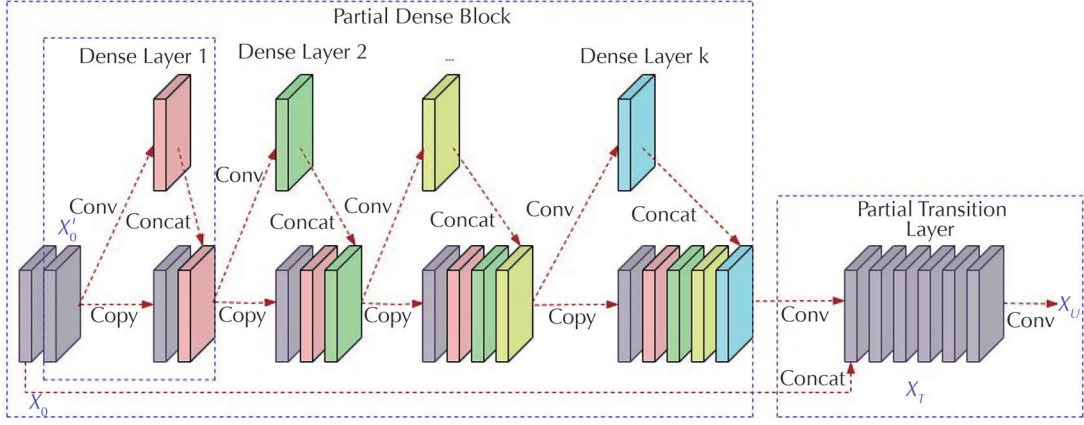


Figure 5. The structure of CSPDenseNet.

The CSPStage [12] is an efficient network module based on channel splitting and cross-stage feature fusion, adhering to two key design principles: feature channel segmentation and crossstage interaction. It processes each scale of features (P3, P4, P5) independently. By extracting deep semantic information through lightweight convolutions while retaining high-resolution detail features, CSPStage reduces computational redundancy and enhances model representational capabilities, particularly improving the detection performance for multi-scale road objects, and significantly boosting the robustness of small and occluded object detection as well as training stability. As shown in Figure 6, CSPStage divides the input feature X into two paths along the channel dimension: a shortcut path X_1 for shallow feature transmission and a main path X_2 for in-depth feature processing. For X_1 , a 1×1 convolution or identity mapping is applied to generate X'_1 , ensuring the preservation of original feature details. Meanwhile, X_2 undergoes multilevel non-linear transformations through N residual blocks (ResBlock) or convolutional layers to extract high-order semantic features, producing X'_2 . Finally, the outputs of the two paths are concatenated along the channel dimension to complete feature fusion, described by the following equations: to extract high-order semantic features, producing X'_2 . Finally, the outputs of the two paths are concatenated along the channel dimension to complete feature fusion, described by the following equations:

$$X'_1 = \text{Conv}_{1 \times 1}(X_1) \quad (6)$$

$$X'_2 = \text{ResBlock}_N(X_2) \quad (7)$$

$$Y = \text{Concat}(X'_1, X'_2) \quad (8)$$

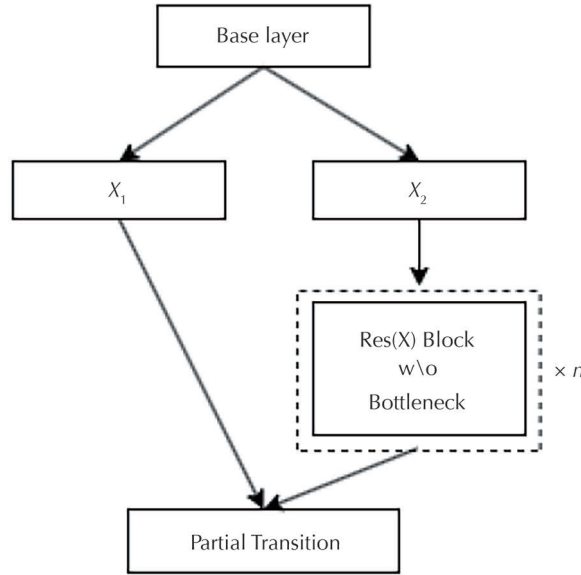


Figure 6. The structure of CSPStage.

During the interaction phase between the model’s Decoder and Head, the IOU-aware query selection module enables dynamic optimization of target queries by calculating the Intersection over Union (IOU) between predicted bounding boxes and ground-truth bounding boxes. Specifically, the module retains high-confidence queries with an IOU no less than 0.5, filters out redundant queries with an IOU below 0.2, and modulates ambiguous queries (with an IOU ranging from 0.2 to 0.5) using the spatial weight matrix W_s output by the EMA module. Furthermore, it collaborates with the EMA and CSPStage modules to establish an end-to-end closed loop of “feature enhancement-purification-query optimization,” thereby further ensuring real-time detection performance in complex scenarios.

4. EXPERIMENTS

This section details the experimental design, implementation, and result analysis. First, we systematically introduce the selection criteria of experimental datasets, the configuration of experimental equipment, and the establishment of evaluation metrics. By conducting experiments on multiple traffic scene datasets, we comprehensively compare the performance of the proposed method with classical algorithms. Subsequently, to analyze the individual contributions of each component in the proposed enhancement method, ablation experiments are performed, followed by a performance comparison with state-of-the-art methods. The experimental results not only validate the effectiveness of our approach but also identify its limitations.

4.1 Experimental Datasets

SODA10M, KITTI, and BDD100K are widely adopted benchmark datasets in the fields of computer vision and autonomous driving, primarily used for algorithm training and performance evaluation in tasks such as object detection, semantic segmentation, and object tracking. These datasets exhibit distinct characteristics in terms of scene coverage, data features, and annotation specifications, providing essential support for comprehensive algorithm assessment.

The SODA10M dataset [14], the primary validation dataset herein, is a large-scale, highchallenge autonomous driving dataset. It covers complex scenarios (rain/snow/fog, day/night, highway/urban/rural roads, occlusions) and includes hard-to-detect elements (small distant/occluded objects), making it ideal for testing cross-environment adaptability and generalization. It provides 5k training, 5k validation, and 10k test images; due to unavailable test annotations, the validation set was used (typical scenarios in Figure 7: highway side views, rainy roads, dense urban traffic).



Figure 7. Part images of SODA10M: Including highways from roadside perspectives, rainy roads, complex urban roads, etc.

The KITTI dataset [15], a classic benchmark, validates algorithms for 3D object detection, stereo vision, and path planning. It integrates camera and LiDAR data, covers urban/highway scenes, and serves as an industry standard via high-precision annotations and standardized protocols.

The BDD100K dataset [16], designed for autonomous driving, supports multi-scenario tasks with more comprehensive annotations (e.g., traffic signs, pedestrian behaviors) and broader scenes than KITTI, challenging model generalization in real-world conditions.

For further generalization verification, KITTI was split into training/validation/test sets at 8:1:1; 10k random images from BDD100K used the same split. Combining these three datasets ensured comprehensive cross-environment performance assessment and reliable experimental results.

4.2 Experimental Setup and Evaluation Metrics

To ensure fair performance comparison between FECS-DETR and baseline models (YOLOv5 [30], YOLOv8 [31], YOLOv9 [4], YOLOv10 [32], YOLOv11 [33], YOLOv12 [34], RT-DETR), all models adopted identical training configurations (excluding structural hyperparameters like backbone depth) via a unified protocol (detailed in Table 1) to isolate architectural impacts. Models were developed with PyTorch on NVIDIA GPUs. For result reliability, each dataset used independent training/validation. The computational settings included a batch size of 4, an input resolution of 640×640, an initial learning rate of 0.0001, and 100 epochs. The experimental environment was configured as follows: Windows 11 operating system, Intel(R) Core (TM) i9-14900KF processor (3.20 GHz), 128 GB of RAM (Intel Corporation, Santa Clara, CA, USA), and an NVIDIA GeForce RTX 4080 SUPER GPU (Nvidia, Santa Clara, CA, USA). The deep learning framework utilized PyTorch 2.5.0, Python 3.9, and CUDA 12.4 for GPU-accelerated computing. For model performance evaluation, an IOU threshold range of 0.5–0.95 was employed to distinguish between foreground and background detections. Specific evaluation metrics included: Recall (R), which measures the model’s ability to detect all ground-truth objects; Precision (P), reflecting the accuracy of detected results; Mean Average Precision (mAP@50) at an IOU threshold of 0.5; and mAP@50:95 (average precision calculated across IOU thresholds from 0.5 to 0.95 with a step of 0.05). Additionally, model complexity

Table 1. Unified Experimental Configuration for All Models.

Parameter Category	Specific Setting
Optimizer	AdamW (weight decay = 0.0001, $\beta_1 = 0.9$, $\beta_2 = 0.999$)
Learning Rate Decay	Cosine annealing (final learning rate = 1e-6)
Data Augmentation	1. Random horizontal flip (probability = 0.5); 2. Color jitter (brightness/contrast/saturation = 0.2); 3. Mosaic augmentation (probability = 0.8); 4. Random crop (scale range = 0.6–1.0)
Loss Function	Focal loss ($\alpha = 0.25$, $\gamma = 2.0$) for classification; L1 loss for bounding box regression
Initialization	All models initialized with pre-trained weights on ImageNet-1K
Gradient Clipping	Max norm = 1.0

and computational efficiency were quantitatively analyzed by measuring the number of parameters (Params) and the number of giga floating-point operations (GFLOPs), enabling a comprehensive assessment of model performance and resource utilization.

4.3 Comparison with State-of-the-Art Models

To systematically evaluate the performance of the FECS-DETR model in practical applications, we selected several leading models in the field of object detection as baselines for comparison. These comparative models represent the forefront of object detection technology, covering multiple technical aspects from speed-accuracy trade-off optimization to enhanced feature extraction capabilities.

4.3.1 Experimental Results on the SODA10M Dataset

To validate the object detection performance of FECS-DETR in complex traffic scenarios, a comprehensive comparative experiment was conducted on the SODA10M dataset. Experimental results, as shown in Table 2, demonstrate that the FECS-DETR series outperforms other mainstream models across multiple key metrics. Specifically, FECS-DETR-18 exhibits excellent parameter utilization efficiency: compared with the baseline RT-DETR-18, its parameter count increases by 24.1% relative to the baseline, while achieving an approximate 2.5% relative improvement in mAP@50. FECS-DETR-50 serves as the optimal trade-off node, with its mAP@50 reaching 0.601 and a frame rate (FPS) maintained at 148.7, which still meets the real-time requirements for autonomous driving scenarios. This performance improvement is primarily attributed to the synergistic optimization architecture composed of FasterBlock, EMA, and CSPStage, which simultaneously enables efficient parameter utilization and enhanced robustness. Furthermore, FECS-DETR-50 only requires a 10.5% relative increase in parameter count to achieve a 4.0% relative improvement in mAP@50. It is worth noting that when compared with large-scale models such as RT-DETR, the FECS-DETR series exhibits significant advantages in detection accuracy. Thus, the proposed model is suitable for application scenarios with strict precision requirements, including autonomous driving and traffic monitoring systems. These experimental results fully highlight the crucial role of the FECS-DETR series in advancing the practical application of intelligent transportation.

Table 2. RESULTS OF DIFFERENT MODELS ON THE SODA10M DATASET.

Model	Params (M)	GFLOPs	mAP@50	mAP@50:95	P	R	FPS (bs = 4)
YOLOv5-S	7.02	15.8	0.503	0.32	0.673	0.445	208.8
YOLOv5-M	20.9	47.9	0.552	0.364	0.652	0.443	186.7
YOLOv5-L	46.1	107.7	0.581	0.388	0.654	0.549	172.4
YOLOv8-S	11.2	28.8	0.482	0.307	0.609	0.440	199.2
YOLOv8-M	25.9	79.3	0.515	0.334	0.668	0.457	188.2
YOLOv8-L	43.6	165.4	0.530	0.348	0.648	0.474	188.6

Table 2. *Continued.*

Model	Params (M)	GFLOPs	mAP@50	mAP@50:95	P	R	FPS (bs = 4)
YOLOv9-S	7.2	26.7	0.473	0.303	0.595	0.432	175.6
YOLOv9-M	20.0	76.5	0.518	0.335	0.647	0.461	154.5
YOLOv10-S	8.0	24.5	0.461	0.202	0.601	0.411	191.0
YOLOv10-M	16.5	63.4	0.487	0.318	0.623	0.435	166.7
YOLOv10-L	25.7	126.4	0.510	0.334	0.634	0.460	154.5
YOLOv11-S	9.5	21.7	0.467	0.299	0.642	0.406	200.0
YOLOv11-M	20.1	68.5	0.499	0.325	0.636	0.456	184.1
YOLOv11-L	25.4	87.6	0.515	0.336	0.667	0.449	154.5
YOLOv12-S	9.2	21.2	0.566	0.290	0.596	0.402	192.3
YOLOv12-M	20.2	67.1	0.507	0.329	0.632	0.449	138.0
YOLOv12-L	26.3	88.6	0.511	0.335	0.608	0.479	121.9
RT-DETR-18	19.9	57.2	0.563	0.374	0.698	0.517	189.8
RT-DETR-34	30.0	87.5	0.576	0.382	0.700	0.520	177.8
RT-DETR-50	42.0	130.3	0.578	0.381	0.700	0.547	172.4
RT-DETR-101	74.8	248.2	0.581	0.375	0.715	0.523	128.2
FECS-DETR-18	24.7	67.5	0.577	0.370	0.722	0.519	173.2
FECS-DETR-34	34.6	95.4	0.589	0.369	0.717	0.526	152.8
FECS-DETR-50	46.4	231.2	0.601	0.402	0.722	0.578	148.7
FECS-DETR-101	65.6	268.5	0.599	0.393	0.732	0.556	106.9

4.3.2 Experimental Results on the KITTI Dataset

To comprehensively validate the perception performance of the FECS-DETR model in autonomous driving scenarios, we conducted a systematic comparative experiment on the KITTI dataset, pitting FECS-DETR against current state-of-the-art models. The detailed results are presented in Table 3. The experimental data reveals that FECS-DETR demonstrates exceptional detection capabilities on the KITTI dataset: compared to the RT-DETR baseline model, it achieves a relative growth rate of 0.6%–1.0% in the mAP@50 metric. Notably, FECS-DETR strikes a balance between high detection accuracy and efficient computational performance, further underscoring its reliability and generalization ability in real-world driving scenarios. Whether navigating complex urban road conditions, high-speed highway environments, or challenges

posed by perspective changes, FECS-DETR consistently delivers stable and outstanding performance. This combination of high-precision detection and real-time computing capabilities positions FECS-DETR as a highly promising foundational model for traffic flow analysis and autonomous driving perception tasks, providing significant support for the advancement of intelligent transportation technologies.

Table 3. RESULTS OF DIFFERENT MODELS ON THE KITTI DATASET.

Model	Params (M)	GFLOPs	mAP@50	mAP@50:95	P	R	FPS (bs = 4)
YOLOv5-M	20.9	47.9	0.921	0.699	0.935	0.852	196.5
YOLOv5-L	46.1	107.7	0.926	0.724	0.939	0.865	177.6
YOLOv8-M	25.9	79.3	0.925	0.719	0.943	0.851	189.5
YOLOv8-L	43.6	165.4	0.936	0.751	0.952	0.871	179.2
YOLOv9-S	7.2	26.7	0.915	0.689	0.914	0.857	198.0
YOLOv9-M	20.0	76.5	0.926	0.732	0.936	0.873	151.9
YOLOv10-M	16.5	63.4	0.920	0.712	0.937	0.856	176.9
YOLOv10-L	25.7	126.4	0.922	0.718	0.925	0.862	145.5
YOLOv11-M	20.1	68.5	0.928	0.722	0.904	0.885	188.3
YOLOv11-M	25.4	87.6	0.927	0.723	0.916	0.883	156.2
YOLOv12-L	26.5	89.7	0.926	0.714	0.938	0.855	169.3
RT-DETR-18	19.9	57.2	0.938	0.712	0.926	0.883	184.6
RT-DETR-34	30.0	87.5	0.931	0.695	0.931	0.863	167.8
RT-DETR-50	42.0	130.3	0.935	0.707	0.922	0.897	142.4
RT-DETR-101	74.8	248.2	0.941	0.711	0.928	0.912	128.2
FECS-DETR-18	24.7	67.5	0.943	0.726	0.933	0.887	179.2
FECS-DETR-34	34.6	95.4	0.940	0.703	0.936	0.893	147.2
FECS-DETR-50	46.4	231.2	0.943	0.725	0.925	0.891	144.2
FECS-DETR-101	65.6	268.5	0.947	0.723	0.939	0.885	112.8

4.3.3 Experimental Results on the BDD100K Dataset

To thoroughly evaluate the adaptability of FECS-DETR in complex autonomous driving scenarios, we carried out comparative experiments on the BDD100K dataset, focusing on the model's detection performance. The experimental results are summarized in Table 4, as an industry-standard dataset

encompassing diverse driving scenarios and environmental conditions, BDD100K, with its vast collection of real-world road images captured by in-vehicle cameras, offers an ideal platform for testing model generalization. The experimental data indicates that FECS-DETR performs remarkably well on this dataset: compared to the corresponding RTDETR baseline models across different backbone configurations, it achieves a notable relative growth rate of 0.7%–2.2% in the mAP@50 metric. This outcome not only highlights the model's robust feature extraction and object recognition capabilities but also demonstrates its adaptability to various environmental changes, ensuring stable detection accuracy in complex traffic scenarios. The exceptional performance of FECS-DETR on the BDD100K dataset further validates its effectiveness and reliability in autonomous driving perception tasks, providing strong experimental evidence for its practical deployment in intelligent transportation systems.

Table 4. RESULTS OF DIFFERENT MODELS ON THE BDD100K DATASET.

Model	Params (M)	GFLOPs	mAP@50	mAP@50:95	P	R	FPS (bs = 4)
YOLOv5-M	25.0	64.0	0.564	0.315	0.674	0.526	193.5
YOLOv5-L	46.1	107.7	0.602	0.332	0.750	0.551	178.6
YOLOv8-M	25.9	79.3	0.570	0.320	0.706	0.524	189.5
YOLOv8-L	43.7	165.7	0.577	0.326	0.672	0.541	179.2
YOLOv9-S	7.2	26.7	0.548	0.307	0.624	0.520	198.0
YOLOv9-M	20.0	76.5	0.567	0.322	0.717	0.519	171.9
YOLOv10-M	16.5	63.4	0.564	0.318	0.652	0.532	176.9
YOLOv10-L	25.7	126.4	0.570	0.323	0.646	0.538	165.5
YOLOv11-M	20.1	68.5	0.570	0.321	0.721	0.517	178.3
YOLOv11-L	25.4	87.6	0.573	0.325	0.702	0.527	156.2
YOLOv12-L	26.5	89.7	0.574	0.325	0.695	0.531	164.2
RT-DETR-18	19.9	57.2	0.584	0.293	0.673	0.557	194.6
RT-DETR-34	30.0	87.5	0.584	0.295	0.661	0.563	177.8
RT-DETR-50	42.0	130.3	0.588	0.297	0.689	0.551	152.4
RT-DETR-101	74.8	248.2	0.594	0.298	0.678	0.563	128.2
FECS-DETR-18	24.7	67.5	0.595	0.297	0.699	0.554	180.2
FECS-DETR-34	34.6	95.4	0.588	0.296	0.687	0.554	157.2
FECS-DETR-50	46.4	231.2	0.601	0.303	0.685	0.570	149.2
FECS-DETR-101	65.6	268.5	0.600	0.303	0.684	0.569	112.8

4.4 Ablation Experiments

To thoroughly investigate the individual contributions of each core module in the FECS-DETR model, systematic ablation experiments were conducted on the FasterBlock, EMA (Efficient Multi-scale Attention), and CSPStage modules. RT-DETR-50 was adopted as the baseline model, and all experiments were performed on the SODA10M dataset under strictly consistent experimental settings. The experimental results (Table 5, where “✓” indicates the activation of the corresponding module) demonstrate that introducing any single module can achieve an increase in mean average precision (mAP) compared to the baseline model without any of these modules. Notably, when the FasterBlock is activated independently, the GFLOPs increase drastically from 130.3 to 238.0, and the number of parameters rises from 42.0 million to 47.7 million. This phenomenon arises from the fact that FasterBlock utilizes partial convolution (with a channel segmentation ratio set to 0.5) and an inverted bottleneck Multi-Layer Perceptron (MLP), wherein the 1×1 convolutional layers adopt a channel expansion factor of 4, which introduces additional convolutional operations and parameter overhead. The CSPStage module achieves parameter reduction while enhancing inference speed, which is of great significance for lightweight model design. Furthermore, a synergistic enhancement effect exists among these modules: the EMA module stabilizes features against noise interference, the FasterBlock refines feature representation capabilities, and the CSPStage preserves multi-scale feature consistency. The integration of all three modules results in a remarkable 4.0% relative growth rate in the mAP@50 metric, which surpasses the cumulative relative growth rates obtained from individual module activations. This superiority is attributed to their hierarchical and complementary interactions.

Table 5. RESULTS OF FECS-DETR ABLATION STUDY EXPERIMENT.

FasterBlock	EMA	CSPStage	Params(M)	GFLOPs	mAP@50	mAP@50:95	FPS
			42.0	130.3	0.578	0.381	172.4
✓			47.7	238.0	0.589	0.387	120.5
	✓		42.2	139.0	0.589	0.388	149.2
		✓	40.6	114.9	0.583	0.389	178.6
✓	✓		47.9	246.7	0.598	0.395	104.2
	✓	✓	40.7	123.5	0.591	0.393	144.9
✓	✓	✓	48.4	231.2	0.601	0.402	148.7

4.5 Robustness Evaluation

To verify the robustness of the FECS-DETR model under complex environmental perturbations, this study designed three input perturbation experiments on the SODA10M validation dataset. Specifically, Gaussian noise with a mean of zero and a standard deviation of 0.1 was added to simulate noise contamination

that image sensors might encounter in practical applications; a motion blur kernel with size 5 was used to replicate blurring from dynamic object displacement in high-speed driving; and brightness and contrast were adjusted by plus minus 30 percent respectively to simulate extreme light changes (e.g., tunnel entry/exit, sunny-rainy transitions). Using RT-DETR-50 as the baseline, the experiments investigated how the EMA module enhances feature stability. As shown in Table 6, the average mAP@50 drop of FECS-DETR under the three perturbations was 3.0 percent, only 53 percent of RT-DETR-50's 5.7 percent. This advantage stems from the synergy of multiple modules: the EMA attention mechanism suppresses noise interference, the FasterBlock module boosts feature robustness, and the CSPStage module maintains multi-scale feature stability. Overall, experiments demonstrate that FECS-DETR significantly improves detection reliability in complex environments via module collaboration. This is crucial for autonomous driving in extreme scenarios (e.g., rain, fog, tunnels), as it enables the model to more stably handle variable environmental challenges in real traffic and supports the safety and reliability of intelligent driving systems.

Table 6. Results of FECS-DETR and RT-DETR under Different Perturbations.

Model	Perturbation Type	Original mAP@50	Post-perturbation mAP@50	Decline Rate
RT-DETR-50	Gaussian noise ($\sigma = 0.1$)	0.578	0.549	4.9%
	Motion blur ($k = 5$)	0.578	0.536	7.3%
	Brightness/contrast $\pm 30\%$	0.578	0.550	4.8%
FECS-DETR-50	Gaussian noise ($\sigma = 0.1$)	0.601	0.583	3%
	Motion blur ($k = 5$)	0.601	0.579	3.7%
	Brightness/contrast $\pm 30\%$	0.601	0.587	2.3%

4.6 Visual Analysis

To intuitively demonstrate the perception capability of the FECS-DETR model in complex road environments, we conducted visualization analysis on its detection results from the SODA10M dataset. Meanwhile, to further validate the model's practical performance, we compared the detection results of FECS-DETR-50 with those of current mainstream object detection models (including YOLOv5-L, YOLOv8-L, YOLOv11-L, and RT-DETR-50) across multiple typical challenging scenarios, focusing on the performance differences in detecting occluded objects and small objects. The visualization results as shown in Figure 8 clearly illustrate the detection capabilities of different models in complex road scenes. In scenarios involving distant small objects and occlusions (e.g., vehicles and pedestrians), FECS-DETR-50 demonstrates superior object localization accuracy and detail-capturing ability. For example, in nocturnal arterial road, the model accurately detects partially occluded pedestrians and non-motor vehicles, whereas other comparative models exhibit varying degrees of missed detections or false positives. Additionally, for distant vehicle objects on complex roads, FECS-DETR-50 generates detection boxes with higher confidence scores and more precise boundary localization, highlighting its advantages in

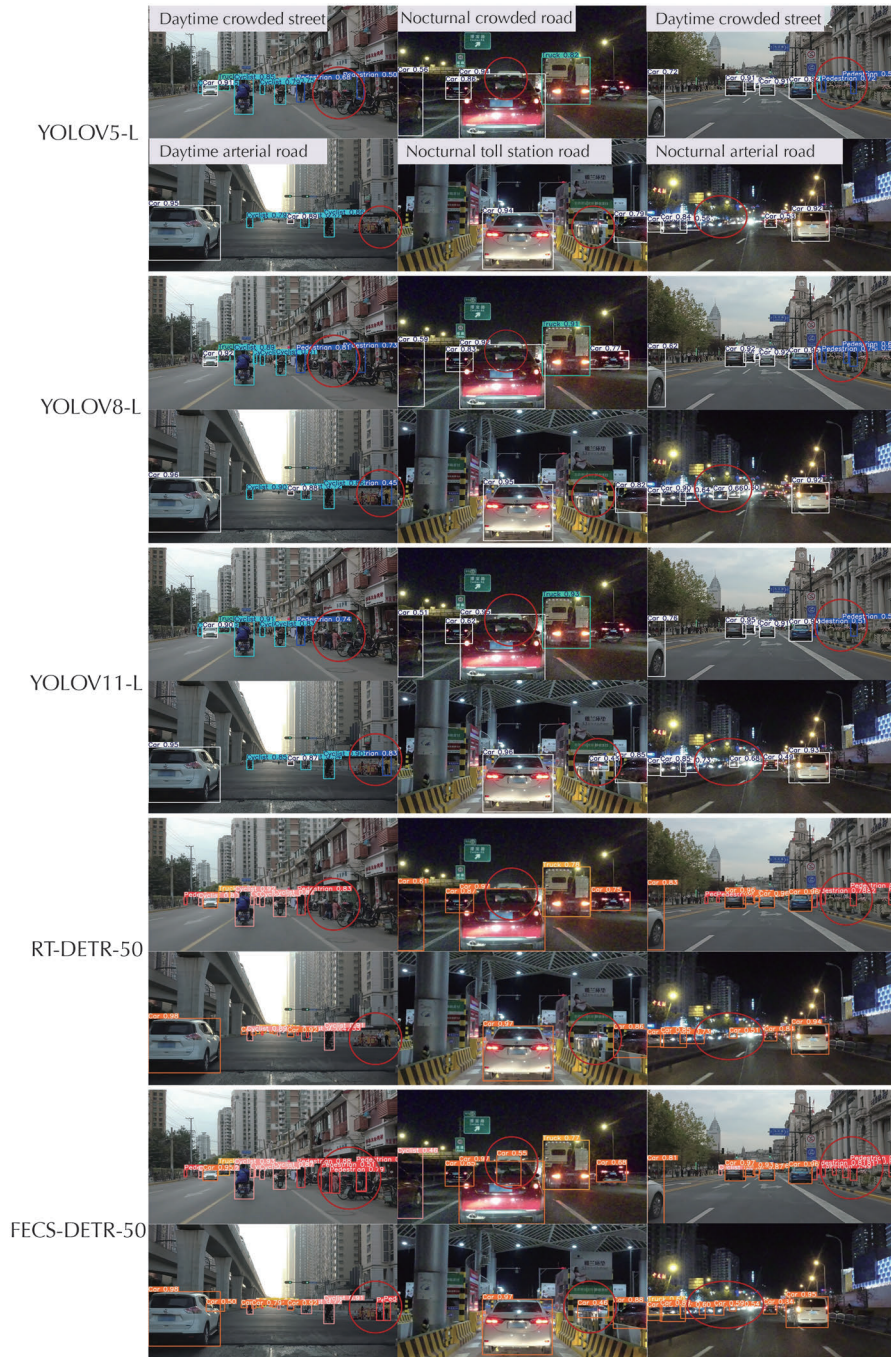


Figure 8. Detection results of overlapping objects in complex road scenes by FECS-DETR-50 compared with classic methods.

multi-scale feature fusion and contextual semantic modeling. Through visual comparisons in real-world scenarios, we directly validate the superiority of the proposed method in addressing complex road perception tasks, particularly in handling detection challenges such as occlusions and small objects. These results further support the effectiveness and reliability of FECS-DETR in practical applications such as autonomous driving and intelligent traffic monitoring.

5. CONCLUSION AND FUTURE WORK

In this paper, the proposed FECS-DETR model ingeniously integrates the FasterBlock, EMA (Efficient Multi-scale Attention), and CSPStage modules to successfully tackle the challenge of detecting small and occluded objects in complex traffic scenarios. The collaborative operation of FasterBlock and EMA enables efficient extraction and retention of shallow texture features and deep semantic information while achieving model lightweighting without compromising critical data. This design ensures the model maintains excellent stability in extreme conditions such as object occlusion and drastic illumination changes. The experimental results are remarkable: on the SODA10M dataset, FECS-DETR surpasses the baseline model by 2.3% in the mAP@50 metric. Additionally, the CSPStage module further optimizes the feature fusion process, effectively suppressing noise interference and enabling the model to accurately detect objects in challenging scenarios such as congested roads and low-light environments. To comprehensively validate the model's effectiveness and generalization ability, we conducted systematic experiments on three representative autonomous driving datasets: SODA10M, KITTI, and BDD100K. The results are gratifying: FECS-DETR not only significantly outperforms existing state-of-the-art models in complex scene detection but also achieves an optimal balance between inference speed and computational efficiency. These achievements provide innovative solutions and robust empirical evidence for the technological development of autonomous driving and intelligent transportation systems. However, the FECS-DETR model still has considerable room for improvement. First, although lightweight design strategies were adopted, the combination of certain modules inevitably increases computational complexity. Future research could explore techniques such as dynamic sparse computing, model pruning, and knowledge distillation to effectively reduce model parameters and computational costs while maintaining detection accuracy. Second, developing efficient hardware-adaptive algorithms to optimize model deployment on edge devices is crucial for enhancing its practical application value. Third, as the current model primarily relies on visual data, integrating multi-source sensor data from millimeter-wave radars, LiDARs, and other sensors represents a promising direction. By fusing multi-modal information, the model's stability in extreme weather conditions (e.g., heavy rain, dense fog) and low-visibility environments can be significantly enhanced, along with improved detection performance for distant and small objects. Finally, incorporating self-supervised pre-training and semi-supervised learning strategies can leverage large volumes of unlabeled data, expand the model's adaptability to long-tail scenarios, reduce dependence on large-scale labeled data, and significantly improve the model's generalization ability. These identified research directions provide a clear roadmap for future work, with the potential to facilitate broader applications of the FECS-DETR model in object detection for complex road scenarios.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

AUTHOR CONTRIBUTIONS

Hong Zhang: Methodology; Data curation; Validation; Writing review and editing; Formal analysis; Visualization.

Linlin Hu: Methodology; Writing original draft; Validation; Writing review and editing; Visualization; Data curation; Conceptualization.

Le Wang: Validation; Data curation.

Yixuan Li: Conceptualization; Data curation.

Xiaoyu He: Conceptualization; Data curation.

ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (62362053), Program for Young Talents of Science and Technology in Universities of Inner Mongolia Autonomous Region of China (NJYT23060) and 2024 Basic Research and Applied Basic Research of Hohhot (2024-G-J-29).

REFERENCES

- [1] J. Mao, S. Shi, X. Wang, and H. Li, "3d object detection for autonomous driving: A comprehensive survey," *International Journal of Computer Vision*, vol. 131, no. 8, pp. 1909–1963, 2023.
- [2] H. Ghahremannezhad, H. Shi, and C. Liu, "Object detection in traffic videos: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 7, pp. 6780–6799, 2023.
- [3] H. Zhang, G. Luo, J. Li, and F.-Y. Wang, "C2fda: Coarse-to-fine domain adaptation for traffic object detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 12633–12647, 2021.
- [4] C.-Y. Wang, I.-H. Yeh, and H.-Y. Mark Liao, "Yolov9: Learning what you want to learn using programmable gradient information," in *European conference on computer vision*. Springer, 2024, pp. 1–21.
- [5] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*. Springer, 2020, pp. 213–229.
- [6] N. Hoanh and T. V. Pham, "Focus-attention approach in optimizing detr for object detection from high-resolution images," *Knowledge-Based Systems*, vol. 296, p. 111939, 2024.
- [7] M. Kothapalli, D. Ravipati, R. Bhatia, "YOLOv1 to YOLOv11: A Comprehensive Survey of Real-Time Object Detection Innovations and Challenges," *arXiv Preprint arXiv:2508.02067*, p. in press, 2025.

- [8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.
- [9] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "Detrs beat yolos on real-time object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 16965–16974.
- [10] J.-W. Liu, D. Yang, T.-W. Feng, and J.-J. Fu, "Md2d-detr: A real-time complex road object detection model based on multi-domain feature decomposition and de-redundancy," *IEEE Transactions on Intelligent Vehicles*, 2024.
- [11] D. Ouyang, S. He, G. Zhang, M. Luo, H. Guo, J. Zhan, and Z. Huang, "Efficient multi-scale attention module with cross-spatial learning," in *ICASSP 2023–2023 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [12] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 390–391.
- [13] J. Chen, S.-H. Kao, H. He, W. Zhuo, S. Wen, C.-H. Lee, and S.-H. G. Chan, "Run, don't walk: chasing higher flops for faster neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 12021–12031.
- [14] J. Han, X. Liang, H. Xu, K. Chen, L. Hong, C. Ye, W. Zhang, Z. Li, X. Liang, and C. Xu, "Soda10m: Towards large-scale object detection benchmark for autonomous driving," 2021.
- [15] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The international journal of robotics research*, vol. 32, no. 11, pp. 1231–1237, 2013.
- [16] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2636–2645.
- [17] A. Haydari and Y. Yilmaz, "Deep reinforcement learning for intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 11–32, 2020.
- [18] H. Zheng, G. Wang, D. Xiao, H. Liu, and X. Hu, "Fta-detr: An efficient and precise fire detection framework based on an end-to-end architecture applicable to embedded platforms," *Expert Systems with Applications*, vol. 248, p. 123394, 2024.
- [19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.
- [21] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.
- [22] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.
- [23] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.
- [24] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

- [25] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.
- [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.
- [27] Y. Chen, C. Zhang, B. Chen, Y. Huang, Y. Sun, C. Wang, X. Fu, Y. Dai, F. Qin, Y. Peng et al, "Accurate leukocyte detection based on deformable-detr and multi-level feature fusion for aiding diagnosis of blood diseases," *Computers in biology and medicine*, vol. 170, p. 107917, 2024.
- [28] H. Zhang, Y. Wang, M. Liang, L. Wang, and E. Hu, "Sand-cyclegan: Physics-constrained synthetic data generation for robust drone object detection in sandstorms," *Data Intelligence*, p. in press, 2025.
- [29] J.-H. Hur, S. Kwon, and H. Kim, "Denoise yourself: Self-supervised point cloud upsampling with pretrained denoising," *Expert Systems with Applications*, vol. 271, p. 126638, 2025.
- [30] Y. Zhang, Z. Guo, J. Wu, Y. Tian, H. Tang, and X. Guo, "Real-time vehicle detection based on improved yolo v5," *Sustainability*, vol. 14, no. 19, p. 12274, 2022.
- [31] Y. Li, Q. Fan, H. Huang, Z. Han, and Q. Gu, "A modified yolov8 detection network for uav aerial image recognition," *Drones*, vol. 7, no. 5, p. 304, 2023.
- [32] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han et al, "Yolov10: Real-time end-to-end object detection," *Advances in Neural Information Processing Systems*, vol. 37, pp. 107984–108011, 2024.
- [33] L.-h. He, Y.-z. Zhou, L. Liu, W. Cao, and J.-h. Ma, "Research on object detection and recognition in remote sensing images based on yolov11," *Scientific Reports*, vol. 15, no. 1, p. 14032, 2025.
- [34] Y. Tian, Q. Ye, and D. Doermann, "Yolov12: Attention-centric real-time object detectors," *arXiv preprint arXiv:2502.12524*, 2025.

AUTHOR BIOGRAPHY



Hong Zhang was born in Inner Mongolia, China, in Jan 1981. She earned her Bachelor's (Transportation, 2001–2004) and Master's (Transportation Planning and Management, 2004–2007) from Harbin Institute of Technology's School of Transportation Science and Engineering, and a Doctor's (Management Science and Engineering, 2012–2015, thesis: *Research of Vehicular Ad Hoc Network Based on Complex Network Theory*) from Huazhong University of Science and Technology. Since 2015, she has been a Professor at Inner Mongolia University's Transportation Institution, focusing on Computer Vision, Digital Twins, and Autonomous Driving.



Linlin Hu was born in Zunyi, Guizhou, China, in 2000. She graduated from the School of Vehicle Engineering, Dalian Jiaotong University in 2023 with a bachelor's degree. From September 2024 to July 2027, she is pursuing a Master's degree in Transportation Engineering at Inner Mongolia University. Her research interests include Computer Vision and Autonomous Driving.



Le Wang was born in Inner Mongolia Autonomous Region, China, in December 2001. He received the Bachelor's degree in Transportation Engineering from Central South University of Forestry and Technology, Changsha, Hunan Province, China in 2020. From September 2024 to July 2027, he is pursuing a Master's degree in Transportation Engineering at Inner Mongolia University. His research interests include Computer Vision and Autonomous Driving.



Yixuan Li was born in Inner Mongolia Autonomous Region, China in January, 1999. He received the Bachelor's degree in Vehicle Engineering from Inner Mongolia University of Science and Technology University, Baotou, China, in 2021. From September 2024 to July 2027, he is pursuing a Master's degree in Transportation Engineering at Inner Mongolia University.



Xiaoyu He was born in Shaoguan, Guangdong, China, in 2002. He received the Bachelor's degree from the College of Automobile and Traffic Engineering. From September 2024 to July 2027, he is pursuing a Master's degree in Transportation Engineering at Inner Mongolia University. His research interests include Computer Vision and Autonomous Driving.