FSR-RTDETR: Frequency-Selective Adaptive Sampling and Structural Re-parameterization for Efficient Traffic Sign Detection

Peiyao Ma, Mengge Lu, Bingjun Liu, Gang Shi*, and Qing Cui*
School of Computer Science and Technology, Xinjiang University, Urumqi, China
107552403972@stu.xju.edu.cn, 107552403966@stu.xju.edu.cn, liubingjun@stu.xju.edu.cn,
shigang@xju.edu.cn, cuiqing@xju.edu.cn

***Abstract*—In real-time traffic sign detection for complex road scenes, background texture interference and cross-scale fusion inconsistency make it difficult to achieve high accuracy and efficiency simultaneously. To tackle this issue, we propose FSR-RTDETR, a real-time detection network built upon RT-DETR. Specifically, we design Frequency-Selective Adaptive Sampling (FSAS) to decouple feature enhancement into frequency-domain suppression and spatial-domain alignment, which mitigates redundant texture responses and strengthens structural representations. We further introduce Learnable Gated Fusion (LGF) to adaptively balance deep semantic cues and shallow fine-grained details, thereby improving cross-scale representation consistency. In addition, we incorporate a Re-parameterized Multi-Branch Convolution (RMBConv) module, which is folded into an equivalent single-branch operator at deployment to avoid any additional inference overhead. Extensive experiments and ablation studies on large-scale datasets validate the effectiveness of the proposed approach.**

***Index Terms*—Traffic sign detection, RT-DETR, Frequency-selective modeling, Multi-scale feature fusion, Structural re-parameterization**

I. Introduction

As pivotal carriers of structured visual information in road traffic scenes, traffic signs provide high-value priors for intelligent driving, vehicle–infrastructure cooperation, and urban traffic management [1]. These priors include speed limits, prohibitions, mandatory instructions, and warnings. However, in complex backgrounds, multi-scale traffic signs make it challenging to simultaneously achieve high detection accuracy and real-time inference efficiency [2]. Previous studies generally followed generic object detection paradigms, falling into two main categories. Two-stage detectors [3], exemplified by Sparse R-CNN [4], focus on proposal generation and refinement. While they mitigate computational redundancy via sparse proposals, their complex inference pipelines often hinder deployment under strict real-time constraints. Conversely, one-stage detectors aim for higher throughput through dense prediction and lightweight designs [5] [6], with models like YOLOv6s [7], YOLOv7-Tiny [8], and YOLOv8n [9] seeing extensive industrial use. However, limited by the intrinsic local receptive fields of convolutional operators, these CNN-based [10] [11] architectures tend to suffer from the attenuation of fine-grained textures and edge details.

Recent studies have increasingly favored task-specific designs emphasizing target enhancement, interference-robust representations, and lightweight design [12]. Notable examples include CABNet [13], which incorporates a context-aware module to improve the discriminative power of the target feature representations. TRD-YOLO [14] integrates global modeling with lightweight heads. TSR-YOLO [15] adopts a multi-stage self-processing strategy to boost the recognizability of tiny signs. YOLO-ADual [16] enhances mobile adaptability through specialized lightweight downsampling and feature extraction modules, and YOLO-DPDG [17] strengthens target representation via mechanisms such as dual pooling and dynamic grouping. Beyond the dominant CNN-based approaches, Transformers [18] are being progressively introduced into traffic sign detection to mitigate the limitations of local convolutions by leveraging their global modeling capabilities. For example, Zeng et al. proposed Transformer-Fusion [19], which employs a cross-scale fusion strategy and reinforced loss constraints to improve the consistency and discriminability of fused representations. Similarly, SE-DETR [20] utilizes lightweight multi-scale feature aggregation and global context modeling to boost detection accuracy and robustness. However, in complex road scenarios, detection performance remains constrained by high-frequency background texture [21] interference and insufficient cross-scale alignment. Consequently, improving cross-scale consistency and representation while maintaining an optimal speed–accuracy balance [22] [23] without increasing inference overhead remains a critical challenge.

Therefore, we propose FSR-RTDETR, an efficient end-to-end detection network based on frequency-selective modeling and structural re-parameterization. The framework constructs a joint frequency–spatial enhancement mechanism via Frequency-Selective Adaptive Sampling (FSAS). This approach improves the discriminability of traffic signs in complex backgrounds through explicit frequency-

Gang Shi* and Qing Cui* are the corresponding authors. This work was supported by the Key Research and Development Project in Xinjiang Uygur Autonomous Region (No.2022B01006).

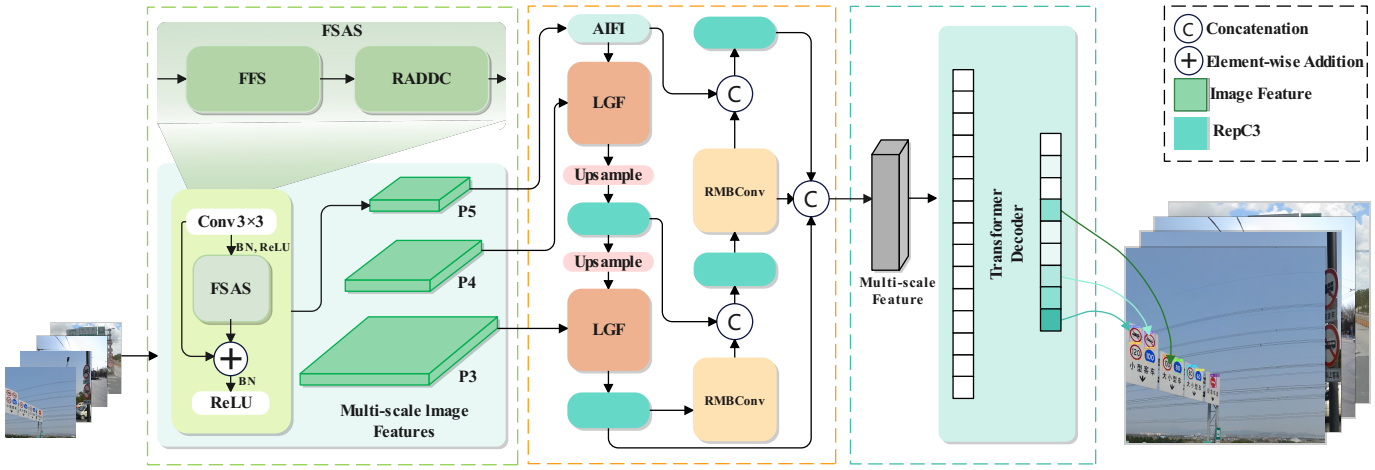

Fig. 1: FSR-RTDETR extracts multi-scale features with backbone, enhances backbone representations via FSAS, performs adjacent-level fusion using LGF, and adopts RMBConv for deployment-friendly convolution.

band selection and radial geometric constraints. Specifically, Frequency-aware Feature Selection (FFS) suppresses redundant background-induced frequency responses while highlighting key structural. Radial Adaptive Dilated Deformable Convolution (RADDC) further regularizes the sampling distribution using a radial sampling strategy aligned with traffic-sign geometric priors. At the multi-scale fusion stage, we introduce Learnable Gated Fusion (LGF), which uses a global learnable gate to adaptively balance deep semantic cues and shallow fine-grained details, combined with lightweight inter-group interactions. And it promotes consistent cross-scale fusion. Finally, we incorporate Re-parameterized Multi-Branch Convolution (RMBConv) as a convolutional building block: during training, its multi-branch topology enhances representational capacity and stabilizes gradient propagation, while at deployment it is folded into an equivalent single-branch operator to enable deployment-friendly inference without additional overhead.

The main contributions of this paper are highlighted as follows:

- We propose FSR-RTDETR, an end-to-end real-time traffic sign detector for complex road scenes, improving the accuracy–efficiency trade-off of RT-DETR.
- We design FSAS, combining frequency-domain denoising with radially geometry-consistent sampling to enhance structural representations.
- We present LGF, a globally gated fusion with lightweight interactions to improve cross-scale consistency and semantic alignment.
- We adopt re-parameterized RMBConv, using multi-branch training and single-branch deployment to avoid extra inference overhead.

Our source code can be found at the following anonymized link: https://anonymous.4open.science/r/FSR-RTDETR-176E.

II. The Proposed Method

A. Overview

As illustrated in Fig. 1, the proposed architecture builds upon RT-DETR [24]. First, the input image is fed into a ResNet-18 backbone for multi-scale feature extraction at the P5 layer. We introduce the Frequency-Selective Adaptive Sampling (FSAS) module to perform frequency-domain information selection and spatial adaptive sampling. This design suppresses background noise while enhancing adaptability to scale variations and geometric deformations. Next, the Learnable Gated Fusion (LGF) module is employed to align and fuse features from adjacent scales via a gating mechanism, thereby mitigating semantic misalignment in multi-scale fusion and ensuring effective feature complementarity. We introduce the Re-parameterized Multi-Branch Convolution (RMBConv) convolution to enhance representational capacity during the training phase while avoiding additional inference overhead. Finally, the fused features are fed into the Transformer encoder-decoder and query prediction heads to output classification and bounding box results, achieving end-to-end real-time detection.

B. Frequency-Selective Adaptive Sampling

In traffic sign detection, the prevalence of high-frequency textural noise frequently interferes with feature discrimination, thereby degrading detection accuracy. To address this challenge, this paper proposes FSAS, a framework incorporating a two-stage decoupled design that prioritizes frequency-domain selection followed by spatial-domain alignment. Specifically, the proposed method comprises Frequency-aware Feature Selection (FFS) and Radial Adaptive Dilation Deformable Convolution (RADDC).

a) Frequency-aware Feature Selection: FFS is not equivalent to simple low-pass denoising. Instead, it performs spectrum-aware spatial re-weighting. As illustrated in Fig. 2, we explicitly model frequency bands using multi-scale low-pass bases and adjacent differential residuals.

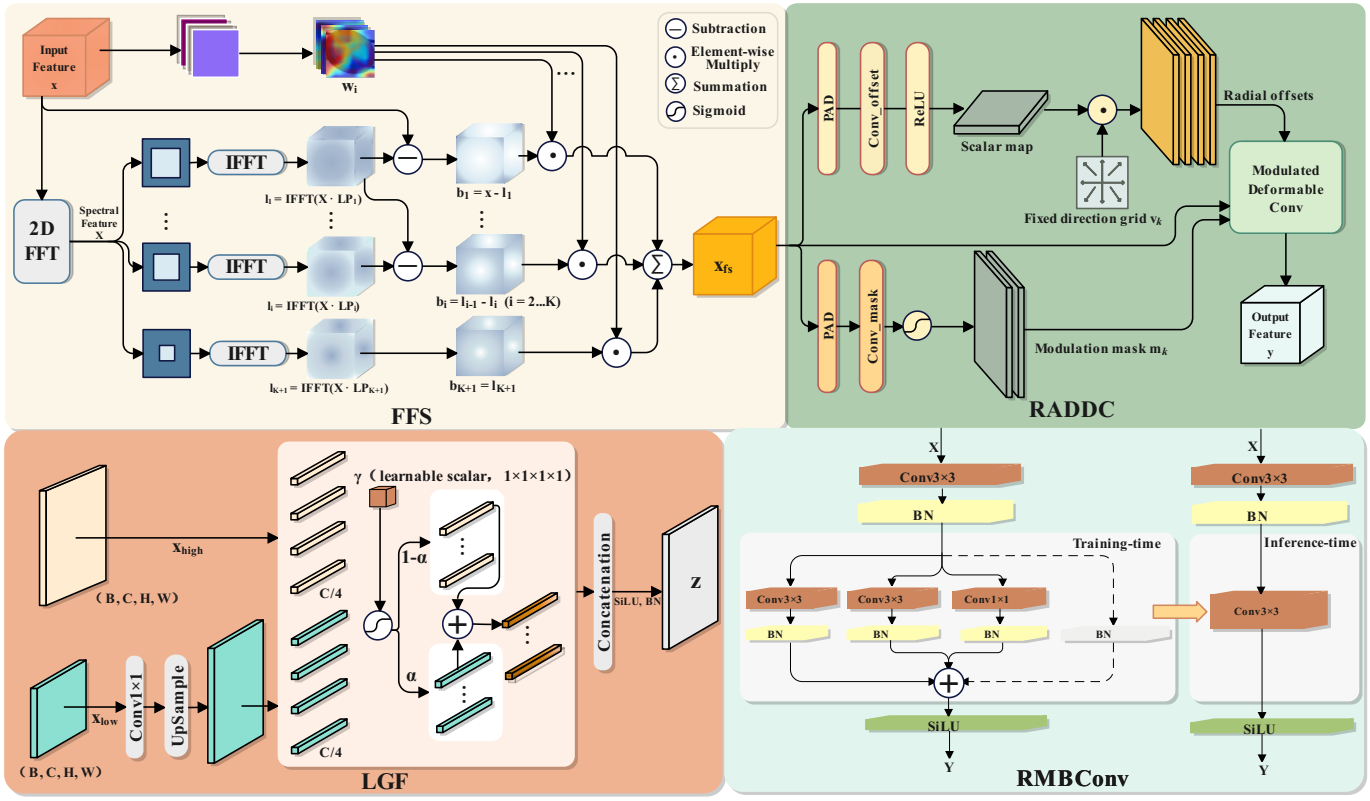

Fig. 2: Architectures of the proposed components: FSAS, which consists of FFS and RADDC, as well as LGF and RMBConv.

Given an input feature map $\mathbf{x} \in \mathbb{R}^{C\times H\times W}$, FFS constructs a set of band-wise components $\{\mathbf{b}_i\}_{i=1}^{K+1}$, where $\mathbf{b}_{K+1}$ denotes the lowest-frequency base term and $\mathbf{b}_i$ $(i = 1, \ldots, K)$ represent residual components associated with higher-frequency structures. In our implementation, each low-pass basis LP_i is an ideal binary mask applied in the Fourier domain, whose centered rectangular passband spans $(H/f_i) \times (W/f_i)$ frequency bins. Then we set $f_i \in \{3, 5, 7, 9\}$ and $K = 4$. To adaptively regulate the contribution of each band at every spatial location, FFS predicts a spatial weight map $\mathbf{w}_i$ for each band via lightweight gating predictors. Concretely, the per-band gating maps are computed as:

$$\mathbf{w}_i = \mathrm{sp\_act}\big(\mathrm{Conv}_i(\mathbf{x})\big), \quad \mathbf{w}_{K+1} = \mathrm{Conv}_{K+1}(\mathbf{x}) \tag{1}$$

where $\mathrm{Conv}_i(\cdot)$ denotes the i-th gating predictor and $\mathrm{sp\_act}(u) = 2\sigma(u)$ produces non-negative, bounded weights. We further adopt a zero-initialization scheme for the gating branch, yielding an approximately identity modulation in early training and thereby stabilizing optimization. After obtaining the band-wise weights, FFS aggregates the band components through element-wise modulation and summation:

$$\mathrm{x}_{fs} = \sum_{i=1}^{K} (\mathbf{w}_i \odot \mathbf{b}_i) + \mathbf{w}_{K+1} \odot \mathbf{b}_{K+1} \tag{2}$$

where $\mathbf{w}_i$ acts as a content-adaptive spatial mask that attenuates spurious activations dominated by high-frequency background textures while retaining structurally informative patterns. Meanwhile, the base term $\mathbf{b}_{K+1}$ provides a stable low-frequency representation, which helps preserve semantic integrity and yields a cleaner feature x_{fs} for subsequent alignment and multi-scale fusion.

b) Radial Adaptive Dilation Deformable Convolution: Upon obtaining the frequency-suppressed feature x_{fs}, RADDC introduces a geometric alignment mechanism. Let $\mathbf{p}_0$ denote the current spatial location on the output feature map. As illustrated in Fig. 2, in contrast to standard deformable convolutions [25] that directly regress unconstrained two-dimensional offsets, RADDC formulates the offsets using fixed grid-aligned directions (we use the canonical 3×3 lattice with $N = 9$ sampling points, denoted as $\{\mathbf{v}_k\}_{k=1}^{9}$, i.e., $\mathbf{v}_k \in \{(-1,-1), (-1,0), \ldots, (1,1)\}$) and learned position-dependent radial scaling factors, together with a modulation mask for reliability-aware aggregation.

Concretely, we predict a scalar map from x_{fs} via an offset predictor, where $\mathrm{PAD}(\cdot)$ denotes appropriate padding for boundary-safe sampling. The scaling map is constrained to be non-negative by $\mathrm{ReLU}(\cdot)$ and is further modulated by a preset coefficient d, yielding:

$$s(\mathbf{p}_0) = \mathrm{ReLU}(\mathrm{Conv}_{\mathrm{off}}(\mathrm{PAD}(\mathrm{x}_{fs}))) \cdot d. \tag{3}$$

Here, d is a layer-wise preset dilation coefficient used to calibrate the predicted scaling response to an offset magnitude

consistent with the target scale, thereby stabilizing training across feature levels. Meanwhile, a mask predictor produces the modulation mask $m_k \in (0,1)$, where k indexes the sampling locations, to down-weight unreliable samples:

$$m_k = \sigma(\mathrm{Conv}_{\mathrm{mask}}(\mathrm{PAD}(\mathrm{x}_{fs}))) \,. \tag{4}$$

The scalar factor $s(\mathbf{p}_0)$ is applied to the preset direction vectors to construct the 2D offset field for deformable sampling. Finally, alignment aggregation is performed via modulated deformable convolution, yielding a cleaner and more spatially aligned representation for subsequent multi-scale fusion and decoding:

$$y = \mathrm{DCN}(\mathrm{x}_{fs},\, s(\mathbf{p}_0),\, m_k) \,. \tag{5}$$

C. Learnable Gated Fusion

Instead of the baseline concatenation-based fusion (i.e., $\mathrm{Concat}([\hat{\mathbf{x}}_{\mathrm{low}},\, \mathbf{x}_{\mathrm{high}}])$ with subsequent convolutional mixing), we adopt LGF to gate the contribution of adjacent-scale features, thereby reducing semantic–detail mismatch and alleviating missed detections. Specifically, given input features from adjacent levels, $\mathbf{x}_{\mathrm{high}}$ provides semantic information, while $\mathbf{x}_{\mathrm{low}}$ preserves fine-grained spatial details. We first obtain the scale-aligned feature $\hat{\mathbf{x}}_{\mathrm{low}}$ by aligning $\mathbf{x}_{\mathrm{low}}$ with $\mathbf{x}_{\mathrm{high}}$, and subsequently partition both features uniformly along the channel dimension into $G = 4$ subgroups for independent fusion. A globally shared learnable scalar, denoted as γ, is mapped via a Sigmoid function to generate a gating coefficient $\alpha = \sigma(\gamma)$. Formally, for the g-th channel group, LGF performs a gated interpolation between the aligned low-level feature $\hat{\mathbf{x}}_{\mathrm{low}}^{(g)}$ and the high-level feature $\mathbf{x}_{\mathrm{high}}^{(g)}$ as:

$$\mathbf{o}^{(g)} = \alpha \odot \hat{\mathbf{x}}_{\mathrm{low}}^{(g)} + (1-\alpha) \odot \mathbf{x}_{\mathrm{high}}^{(g)}, \quad g \in \{1,2,3,4\} \tag{6}$$

where $\odot$ denotes element-wise multiplication. The coefficient α is broadcast across spatial locations (and channels within each group), enabling global ratio calibration with negligible computational overhead. Since the two aligned streams mainly differ in their relative emphasis on high-level semantics versus fine-grained details, rather than pixel-wise misalignment, a spatially shared scalar α is sufficient to calibrate their global mixing ratio with minimal degrees of freedom, improving optimization stability. Finally, the fused outputs from all groups are concatenated along the channel dimension to obtain the final fused representation:

$$\mathbf{z} = \mathrm{Concat}\Big(\mathbf{o}^{(1)},\, \mathbf{o}^{(2)},\, \mathbf{o}^{(3)},\, \mathbf{o}^{(4)}\Big) \,. \tag{7}$$

D. Re-parameterized Multi-Branch Convolution

To enhance the representational capacity of the object detection network for complex features without incurring additional inference costs, we design RMBConv as a drop-in replacement for the baseline downsampling convolution in the backbone, inspired by the structural re-parameterization philosophy [26]. During the training phase, the module leverages a multi-branch approach to expand the solution space. Given an input feature $\mathbf{X}$, the output $\mathbf{Y}$ is computed by aggregating dual parallel 3×3 convolutional branches, a 1×1 convolutional branch, and a conditional identity branch (active when $C_{in} = C_{out}$):

$$\mathbf{Y} = \phi\left(\sum_{k=1}^{2} \mathcal{F}_{3\times3}^{(k)}(\mathbf{X}) + \mathcal{F}_{1\times1}(\mathbf{X}) + \mathbb{I}_{\mathrm{id}}\, \mathcal{B}(\mathbf{X})\right) \tag{8}$$

where $\phi(\cdot)$ is instantiated as the SiLU activation in our implementation, $\mathcal{F}(\cdot)$ represents the convolution followed by Batch Normalization (BN), and $\mathcal{B}(\cdot)$ denotes the BN operation on the identity path. $\mathbb{I}_{\mathrm{id}}$ is an indicator that equals 1 if channel dimensions match and 0 otherwise.

During inference, the linear multi-branch structure is analytically fused into an equivalent single 3×3 convolution, enabling efficient deployment without additional computational overhead. To this end, Batch Normalization (BN) is first folded into the preceding convolution of each branch, yielding the merged parameters $\mathbf{W}'$ and $\mathbf{b}'$. The 1×1 kernel is then expanded to a 3×3 kernel via a zero-padding operator $\mathcal{P}(\cdot)$ to ensure kernel-size alignment. In addition, when the identity branch is enabled ($C_{\mathrm{in}} = C_{\mathrm{out}}$), its BN parameters are converted into an equivalent convolutional kernel $\mathbf{W}'_{\mathrm{id}}$ (with the corresponding bias term). Finally, the equivalent kernel $\mathbf{W}_{\mathrm{eq}}$ and bias $\mathbf{b}_{\mathrm{eq}}$ are obtained by linearly aggregating the merged kernels and biases from all branches, yielding the inference-time form

$$\mathbf{Y} = \phi(\mathrm{Conv}_{3\times3}(\mathbf{X}; \mathbf{W}_{eq}, \mathbf{b}_{eq})) \tag{9}$$

where the fused parameters are given by

$$\begin{cases} \mathbf{W}_{eq} = \displaystyle\sum_{k=1}^{2} \mathbf{W}'^{(k)}_{3\times3} + \mathcal{P}(\mathbf{W}'_{1\times1}) + \mathbb{I}_{\mathrm{id}}\, \mathbf{W}'_{\mathrm{id}}, \\ \mathbf{b}_{eq} = \displaystyle\sum_{k=1}^{2} \mathbf{b}'^{(k)} + \mathbf{b}'_{1\times1} + \mathbb{I}_{\mathrm{id}}\, \mathbf{b}'_{\mathrm{id}}. \end{cases} \tag{10}$$

E. Loss Function

During the training of FSR-RTDETR, a set-based supervision strategy is employed for the predictions produced by the detection decoder. Specifically, a one-to-one assignment between predicted queries and ground-truth objects is first established via bipartite matching, and the matched pairs are then optimized by a weighted combination of classification and localization losses. Eventually, the overall loss function is formulated as:

$$\mathcal{L}_{\mathrm{det}} = \lambda_{\mathrm{cls}}\mathcal{L}_{\mathrm{cls}} + \lambda_1\mathcal{L}_1 + \lambda_{\mathrm{giou}}\mathcal{L}_{\mathrm{giou}}. \tag{11}$$

Here, $\mathcal{L}_{\mathrm{cls}}$ denotes a focal-style classification loss (unmatched predictions are supervised as the "no-object" class), $\mathcal{L}_1$ is the ℓ_1 loss on normalized box parameters, and $\mathcal{L}_{\mathrm{giou}}$ is the generalized IoU loss for bounding box regression. λ_{cls}, λ_1 and λ_{giou} are scalar weights balancing the three terms.

III. Experiments

A. Experimental Setup

a) *Datasets:* We evaluate our method on the TT100K (2021) [27] benchmark, a large-scale dataset derived from Tencent Street View comprising over 100k high-resolution (2048×2048) images across diverse scenarios. To mitigate severe class imbalance, we selected a subset of 45 high-frequency categories (samples ≥ 100) for stable evaluation, covering object scales from 8×8 to 400×400 pixels. Strictly adhering to the official protocol, we partition the original training data into a training set (5,191 images) and a validation set (915 images) using an 8.5:1.5 ratio, and evaluate final results on the official test set (3,072 images).

b) *Implementation Details:* Experiments were performed on an NVIDIA A40 GPU using PyTorch 2.0.1 to ensure reproducibility. Input images were resized to 640×640. The training process consisted of 200 epochs with a batch size of 4, optimized by Adam with an initial learning rate of 1×10^{-4}.

c) *Evaluation Metrics:* Model performance is evaluated using standard object detection metrics. Specifically, we report mAP@0.5, mAP@0.75, mAP@0.5:0.95, Precision (P), and Recall (R) to assess detection accuracy and reliability. Additionally, inference throughput is measured in frames per second (FPS), and computational complexity is quantified in GFLOPs, enabling a systematic assessment of the accuracy–efficiency trade-off.

B. Comparison With Other Methods

To validate the effectiveness of the proposed method in complex road scenarios, we benchmark FSR-RTDETR against various representative detectors on the TT100K-45 dataset. As shown in Table I, FSR-RTDETR achieves an mAP@0.5 of 80.46% and reaches 61.34% on the stricter mAP@0.5:0.95 metric. Compared to the two-stage method Sparse R-CNN, our method delivers a substantial speedup of approximately 18 times while maintaining comparable accuracy. In contrast to lightweight YOLO variants such as TSR-YOLO, although our FPS is slightly lower, we achieve a significant gain of nearly 10 percentage points in mAP@0.5, highlighting enhanced detection reliability in complex scenarios.

C. Ablation Study

Based on the RT-DETR-r18 baseline, we conducted ablation studies on the TT100K-45 dataset to verify the effectiveness of each component. As shown in Table II, FSAS alone yields stable performance gains (mAP@0.5 = 79.53%, mAP@0.5:0.95 = 60.78%). Meanwhile, incorporating LGF alone primarily improves performance in high IoU [29] intervals (mAP@0.75 = 73.26%), indicating its capability to enhance cross-scale fusion quality. The combination of these two modules delivers complementary performance gains (mAP@0.5 = 80.27%). Ultimately, the integration of RMBConv achieves the optimal accuracy-efficiency balance. It elevates the comprehensive performance to 80.46% in mAP@0.5 and 61.34% in mAP@0.5:0.95 while maintaining real-time inference, thereby enhancing deployment friendliness without compromising accuracy. To visually corroborate the quantitative results, Fig. 3 compares the baseline and our method against the ground truth, showing tighter localization and fewer false positives in cluttered scenes.

TABLE I: Comparison with representative detectors on TT100K. Metrics are reported in % with two decimals. ↑/↓ indicates higher/lower is better. Accuracy metrics (mAP, P, R) quantify detection performance, while efficiency metrics (GFLOPs) reflect computational cost, respectively. "–" denotes results not reported. The best result in each column is highlighted in **bold**.

Method	mAP@0.5↑	mAP@0.5:0.95↑	P↑	R↑	GFLOPs↓
Sparse R-CNN [4] (2021)	69.50	53.20	–	–	172.00
TOOD [28] (2021)	70.70	50.70	71.20	77.30	125.00
YOLOv6s [7] (2022)	75.60	57.60	–	–	45.30
YOLOv7 [8] (2022)	52.90	39.40	52.10	55.40	103.90
YOLOv8n [9] (2023)	60.50	46.10	52.90	59.90	**8.70**
TSR-YOLO [15] (2023)	72.73	56.57	–	–	–
YOLO-Adual [16] (2024)	70.10	–	71.80	63.50	11.20
YOLO-DPDG [17] (2025)	69.50	52.90	69.80	61.00	9.20
SE-DETR [20] (2025)	79.70	61.00	84.20	76.00	–
Ours	**80.46**	**61.34**	**85.56**	**77.38**	53.50

Fig. 3: Qualitative comparison among (1) ground-truth annotations, (2) the baseline model, and (3) the proposed model.

Overall, FSR-RTDETR achieves a more favorable accuracy-efficiency trade-off in complex road scenes. These results validate the effectiveness of the proposed mechanisms, including frequency-domain noise suppression, geometric-consistent sampling, unified fusion, and the deployment-friendly re-parameterized structure.

D. Effective Receptive Field Analysis

To elucidate the performance gains in feature aggregation, we quantitatively measured the effective receptive field of feature maps (specifically at Layer 19), following the protocol established by RepLKNet [30]. ERF serves as a critical metric for gauging a network's capacity to capture long-range depen-

TABLE II: Ablation study on TT100K. Metrics (mAP, P, R) are reported in % with two decimals. ↑/↓ indicates higher/lower is better. "FPS (infer.)" is measured under the same setting. "Params" denotes the total number of trainable parameters (in M) of the whole detector.

Modules			Accuracy ↑					Efficiency		
FSAS	LGF	RMBConv	mAP@0.5	mAP@0.75	mAP@0.5:0.95	P	R	GFLOPs↓	Params(M)↓	FPS ↑
			79.11	71.50	60.07	81.62	**78.51**	57.10	19.92	138.27
✓			79.53	72.78	60.78	83.59	78.20	57.10	19.92	140.32
	✓		80.32	73.26	61.21	85.11	78.04	57.10	19.93	137.97
✓	✓		80.27	72.23	60.95	85.61	77.59	57.10	19.93	124.01
	✓	✓	80.31	72.88	61.26	**85.97**	76.93	57.10	19.93	**171.16**
✓	✓	✓	**80.46**	**73.28**	**61.34**	85.56	77.38	**53.50**	20.07	154.94

TABLE III: ERF coverage ratio at different contribution thresholds on the P3 branch. Values are in % (higher indicates broader spatial coverage of high-contribution regions).

Model	t=20%	t=30%	t=50%	t=99%
RT-DETR-r18	10.66	18.73	40.44	99.06
Ours	**12.80 (+ 2.14)**	**23.01 (+ 4.28)**	**44.10 (+ 3.66)**	99.06

dencies. As shown in Table III, compared to the Baseline, the Full Model exhibits a significant expansion in coverage. Specifically, at core thresholds of t = 20% and t = 30%, the coverage ratios increase from 10.66% to 12.80% (+ 2.14) and from 18.73% to 23.01% (+ 4.28), respectively. At t = 50%, the coverage expands from 40.44% to 44.10% (+ 3.66). These results indicate that the proposed design effectively expands the spatial coverage of high-contribution regions within the P3 branch, thereby enhancing context utilization for traffic signs against complex backgrounds. Note that the threshold t denotes the cumulative contribution ratio. The percentages in the table represent the ratio of the minimum center-anchored square area required to reach the cumulative contribution t relative to the total input area.

IV. Conclusion

To tackle the challenge of balancing accuracy and efficiency in traffic sign detection under complex road scenarios, this paper proposes FSR-RTDETR, an efficient detection network based on frequency-selective modeling and structural re-parameterization. By leveraging the frequency-spatial decoupling mechanism of the FSAS module and the parameter-sharing gating strategy of the LGF module, we effectively mitigate feature degradation caused by background interference and semantic misalignment during cross-scale fusion, without incurring significant computational overhead. Furthermore, the incorporation of RMBConv guarantees deployment efficiency via structural re-parameterization. Extensive experiments on the TT100K dataset demonstrate that FSR-RTDETR achieves a Precision of 85.56% and an mAP@0.5 of 80.46% while maintaining high real-time inference speeds. Achieving superior results among the compared methods, it provides a solution combining high performance and low latency for autonomous driving perception systems. In future work, we will evaluate FSR-RTDETR on more diverse benchmarks and adverse conditions to further improve its generalization while maintaining real-time efficiency.

References

[1] C. Wang, B. Zheng, and C. Li, "Efficient traffic sign recognition using yolo for intelligent transport systems," *Scientific Reports*, vol. 15, no. 1, p. 13657, 2025.

[2] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *Journal of Field Robotics*, vol. 37, no. 3, pp. 362–386, 2020.

[3] B. Wagner, D. Pellerin, and S. Huet, "Studying forgetting in Faster R-CNN for online object detection: Analysis scenarios, localization in the architecture, and mitigation," *IEEE Access*, vol. 13, pp. 6067–6079, 2025.

[4] P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, Z. Yuan, and P. Luo, "Sparse R-CNN: An end-to-end framework for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 12, pp. 15 650–15 664, 2023.

[5] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," *arXiv preprint arXiv:1612.08242*, 2016.

[6] C. Wang, W. He, Y. Nie, J. Guo, C. Liu, Y. Wang, and K. Han, "Gold-YOLO: Efficient object detector via gather-and-distribute mechanism," *Advances in Neural Information Processing Systems*, vol. 36, pp. 51 094–51 112, 2023.

[7] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie *et al.*, "YOLOv6: A single-stage object detection framework for industrial applications," *arXiv preprint arXiv:2209.02976*, 2022.

[8] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," arXiv preprint arXiv:2207.02696, 2022.

[9] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," GitHub repository, 2023, version 8.0.0. Accessed: 2026-01-26. [Online]. Available: https://github.com/ultralytics/ultralytics

[10] G. More, O. Patil, O. More, M. More, S. Suryavanshi, and M. Mali, "Comparison of object detection algorithms CNN, YOLO and SSD," *International Journal of Scientific Research and Technology*, 2024.

[11] U. Mittal, P. Chawla, and R. Tiwari, "EnsembleNet: A hybrid approach for vehicle detection and estimation of traffic density based on Faster R-CNN and YOLO models," *Neural Computing and Applications*, vol. 35, no. 6, pp. 4755–4774, 2023.

[12] S. Zhang, S. Che, Z. Liu, and X. Zhang, "A real-time and lightweight traffic sign detection method based on Ghost-YOLO," *Multimedia Tools and Applications*, vol. 82, no. 17, pp. 26 063–26 087, 2023.

[13] L. Cui, P. Lv, X. Jiang, Z. Gao, B. Zhou, L. Zhang, L. Shao, and M. Xu, "Context-aware block net for small object detection," *IEEE Transactions on Cybernetics*, vol. 52, no. 4, pp. 2300–2313, 2022.

[14] J. Chu, C. Zhang, M. Yan, H. Zhang, and T. Ge, "TRD-YOLO: A real-time, high-performance small traffic sign detection algorithm," *Sensors*, vol. 23, no. 8, p. 3871, 2023.

[15] W. Song and S. A. Suandi, "TSR-YOLO: A chinese traffic sign recognition algorithm for intelligent vehicles in complex scenes," *Sensors*, vol. 23, no. 2, p. 749, 2023.

[16] S. Fang, C. Chen, Z. Li, M. Zhou, and R. Wei, "YOLO-ADual: A lightweight traffic sign detection model for a mobile driving system," *World Electric Vehicle Journal*, vol. 15, no. 7, p. 323, 2024.

[17] R. Liang, M. Jiang, and S. Li, "YOLO-DPDG: A dual-pooling dynamic grouping network for small and long-distance traffic sign detection," *Applied Sciences*, vol. 15, no. 20, p. 10921, 2025.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," arXiv preprint arXiv:1706.03762, 2017.

[19] G. Zeng, W. Huang, Y. Wang, X. Wang, and W. E, "Transformer fusion and residual learning group classifier loss for long-tailed traffic sign detection," *IEEE Sensors Journal*, vol. 24, no. 7, pp. 10 551–10 560, 2024.

[20] R. Zhang and M. Zhang, "Research on traffic sign detection algorithm using enhanced real-time detection transformer," in *2025 Joint International Conference on Automation-Intelligence-Safety & International Symposium on Autonomous Systems*, 2025, pp. 1–6.

[21] Y. Shen, K. Peng, L. Liu, W. Ji, J. Li, M. Zhang, Y. Piao, and H. Lu, "Frequency-domain decomposition and recomposition for robust audio-visual segmentation," arXiv preprint arXiv:2509.18912, 2025.

[22] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 781–10 790.

[23] Z. Li, H. Chen, B. Biggio, Y. He, H. Cai, F. Roli, and L. Xie, "Toward effective traffic sign detection via two-stage fusion neural networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 8, pp. 8283–8294, 2024.

[24] 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," arXiv preprint arXiv:2304.08069, 2024.

[25] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *2017 IEEE International Conference on Computer Vision*, 2017, pp. 764–773.

[26] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun, "RepVGG: Making VGG-style ConvNets great again," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 728–13 737.

[27] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2110–2118.

[28] C. Feng, Y. Zhong, Y. Gao, M. R. Scott, and W. Huang, "TOOD: Task-aligned one-stage object detection," in *2021 IEEE/CVF International Conference on Computer Vision*, 2021, pp. 3490–3499.

[29] Y.-F. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, and T. Tan, "Focal and efficient IoU loss for accurate bounding box regression," arXiv preprint arXiv:2101.08158, 2022.

[30] X. Ding, X. Zhang, Y. Zhou, J. Han, G. Ding, and J. Sun, "Scaling up your kernels to 31x31: Revisiting large kernel design in CNNs," arXiv preprint arXiv:2203.06717, 2022.