applied sciences

MDPI

Article

RD-DETR: A Robust Vehicle Detector via Reaction–Diffusion Mechanisms

Yi Huang [1], Yishi Chen [2], Kaiming Pan [1], Xiangning Wu [2], Haoxiang Huang [1] and Yanmei Meng [1,*]

[1] School of Mechanical Engineering, Guangxi University, Nanning 530004, China; 2311391033@st.gxu.edu.cn (Y.H.)
[2] Guangxi Research Institute of Mechanical Industry Co., Ltd., Nanning 530007, China
* Correspondence: mengtang@gxu.edu.cn

Abstract

Vehicle detection is a fundamental perception task in intelligent transportation systems and autonomous driving. Although state-of-the-art detectors achieve competitive performance under normal conditions, their robustness degrades substantially under adverse conditions such as rain, fog, low illumination, and sensor noise. To address this challenge, we propose RD-DETR, a vehicle detector that incorporates reaction–diffusion mechanisms into deep feature learning. The RDNet backbone adopts a pyramid-based enhancement strategy in which shallow layers preserve fine-grained texture details while deep layers employ reaction–diffusion-inspired dynamics to suppress noise and enhance target representations. The Phase-Guided Spatial Attention (PGSA) module leverages phase-related structural cues that are relatively less sensitive to global illumination and contrast variations, helping recover vehicle boundaries when appearance cues become unreliable under adverse imaging conditions. The Content-Aware Adaptive Fusion Module (CA-AFM) dynamically aggregates multi-scale features according to scene complexity, improving detection across diverse traffic scenarios. Experiments on BDD100K and DAWN show that RD-DETR yields mAP@0.5 improvements of 3.2 and 4.0 percentage points over RT-DETR, respectively, while reducing model parameters by 27.6%, indicating a favorable balance between accuracy and efficiency under the evaluated settings.

Keywords: vehicle detection; physics-inspired neural networks; reaction–diffusion equations; phase-guided attention; adaptive feature fusion

1. Introduction

Vehicle detection is an essential perception task in intelligent transportation systems (ITSs) and autonomous driving. In ITS applications, accurate vehicle detection underpins traffic flow monitoring, adaptive signal control, and intelligent law enforcement. In autonomous driving, reliable detection is a prerequisite for path planning and collision avoidance, with direct implications for driving safety.

The advent of deep learning has driven substantial progress in vehicle detection. From two-stage detectors [1,2] to efficient single-stage approaches [3–8] and Transformer-based end-to-end frameworks [9,10], both accuracy and inference speed have improved considerably. However, existing detectors exhibit insufficient robustness under complex degradation conditions, such as rain, fog, low illumination, and sensor noise. In practical deployment, algorithm design remains constrained by trade-offs among inference speed, environmental adaptability, and computational cost.

Check for updates

Received: 22 March 2026
Revised: 25 April 2026
Accepted: 27 April 2026
Published: 30 April 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Three fundamental challenges persist for vehicle detection under degraded conditions. First, diverse degradation patterns coexist in complex scenes: atmospheric scattering blurs fine details, rain and snow streaks inject high-frequency noise, low-light conditions reduce signal-to-noise ratios, and motion blur coupled with sensor noise further degrades image quality. Existing methods, whether CNN-based or Transformer-based, rely on data-driven statistical learning and lack explicit modeling of physical imaging processes, making it difficult to separate targets from background noise under severe degradation. Although physics-based modeling has demonstrated value in low-level vision, including deep denoising [11] and trainable reaction–diffusion models [12], such approaches remain underexplored in real-time object detection. Second, existing attention mechanisms predominantly operate in spatial and channel domains. While some works have explored global frequency features [13], they focus primarily on magnitude information. Prior signal processing studies suggest that phase information captures structural cues and is often less sensitive to global illumination and contrast changes than magnitude information [14], which may help preserve boundary contours in degraded images; however, this cue remains underutilized in detection. Third, traffic scenarios span a wide scale range, yet existing multi-scale fusion strategies such as FPN [15] and BiFPN [16] employ fixed weights and cannot adapt dynamically to image content, limiting their effectiveness under complex degradation.

To address these challenges, this study aims to develop an end-to-end multi-class detector that remains both accurate and computationally efficient under adverse traffic imaging conditions. In this work, robustness refers to detection stability under imaging degradations, such as low illumination, adverse weather, blur, and sensor noise, rather than to all possible challenges in traffic-scene detection. The investigation is structured around three research questions: RQ1: Can embedding reaction–diffusion-inspired dynamics into the backbone improve the detector's accuracy–efficiency trade-off under adverse imaging conditions? RQ2: Can phase-guided cues provide complementary structural information when appearance cues become unreliable? RQ3: Can content-aware adaptive fusion improve multi-scale aggregation relative to static fusion strategies? Building on these questions, we propose RD-DETR, whose three main contributions are summarized below.

The main contributions of this work are as follows:

1. We propose RDNet, a backbone that integrates reaction–diffusion mechanisms into feature extraction. A pyramid-based enhancement strategy is adopted, preserving textural details in shallow layers while suppressing noise and enhancing target representations through reaction–diffusion dynamics in deep layers.
2. We design the Phase-Guided Spatial Attention (PGSA) module, which leverages phase-related structural cues in the frequency domain to guide spatial attention, enabling improved extraction of vehicle contours under low signal-to-noise conditions.
3. We introduce the Content-Aware Adaptive Fusion Module (CA-AFM), which dynamically generates scale-specific weights through a lightweight network, improving multi-scale detection accuracy while remaining suitable for resource-constrained deployment.

Experiments on BDD100K and DAWN show that RD-DETR yields mAP@0.5 improvements of 3.2 and 4.0 percentage points over RT-DETR, respectively, while reducing model parameters by 27.6%, indicating a favorable balance between accuracy and efficiency under the evaluated settings.

The remainder of this paper is organized as follows. Section 2 reviews related work on real-time object detection, robust detection under adverse conditions, and physics-inspired deep learning. Section 3 presents the RD-DETR architecture, including RDNet, PGSA, and CA-AFM. Section 4 reports the experimental setup, comparative results, targeted ablation studies, repeated-run statistics, and visualization analysis. Section 5 concludes this

paper, summarizes how the research questions are addressed, and discusses limitations and future directions.

2. Related Work

2.1. Real-Time Object Detection

Object detection has evolved from two-stage to single-stage architectures and subsequently to Transformer-based paradigms. Early region proposal network methods, including Faster R-CNN [1] and Cascade R-CNN [2], achieved high detection precision but were limited in inference speed. Single-stage detectors subsequently dominated real-time applications. RetinaNet [3] addressed class imbalance through Focal Loss while preserving efficiency. The YOLO series [5–8] has continuously advanced performance through diverse architectural innovations: YOLOv9 introduced programmable gradient information for improved learning, YOLOv10 achieved NMS-free end-to-end training, YOLOv12 incorporated attention mechanism optimization, and YOLOv13 proposed hypergraph-enhanced adaptive visual perception, collectively pushing the accuracy–speed frontier. The introduction of Transformer architectures brought a new paradigm to object detection. DETR [9] realized end-to-end detection through self-attention mechanisms and the Hungarian matching algorithm, eliminating dependence on anchors and non-maximum suppression. RT-DETR [10] addressed real-time detection requirements by designing an efficient hybrid encoder and an uncertainty-minimal query selection strategy, achieving a competitive accuracy–speed trade-off. Lightweight backbone networks such as FasterNet [17], StarNet [18], and MobileNetV4 [19] have further reduced computational overhead while maintaining performance, providing viable options for edge deployment. Nevertheless, these models lack explicit modeling of physical imaging processes, limiting their robustness under complex degradation conditions.

2.2. Robust Detection Under Challenging Conditions

Considerable research has addressed detection degradation caused by adverse image quality. One line of work enhances robustness through architectural design: Kang et al. [20] proposed YOLO-FA, which dynamically reweights feature maps via fuzzy entropy to reduce uncertainty in rainy and nighttime scenarios; Liu et al. [21] designed MDFD2-DETR, employing feature decomposition across spatial, channel, and frequency domains to handle complex scenes; and Tahir et al. [22] integrated Swin–Transformer attention into YOLOv8 to improve small object recognition. Another line adopts image preprocessing, such as applying ESRGAN for enhancement prior to detection [23]. Although effective in specific scenarios, preprocessing-based methods introduce additional computational overhead and are not amenable to end-to-end optimization, limiting their practicality in real-time systems. Across both lines, explicit modeling of the physical processes underlying image degradation remains largely absent.

2.3. Physics-Inspired Deep Learning

Physics-inspired methods have demonstrated strong potential in low-level vision. Classical anisotropic diffusion achieves a favorable balance between noise suppression and edge preservation through adaptive diffusion coefficients. Chen and Pock [12] proposed TNRD, a trainable nonlinear reaction–diffusion framework that learns system parameters in a data-driven manner, achieving strong performance in image denoising and super-resolution. Reaction–diffusion systems can simultaneously achieve noise smoothing and feature enhancement through the coupled action of diffusion and reaction terms. However, such physics-inspired approaches have primarily been applied to low-level tasks and remain underexplored in high-level vision tasks such as real-time object detection, where

they are typically relegated to preprocessing or training-time regularization. The present work addresses this gap by embedding reaction–diffusion-inspired mechanisms directly into backbone network design, modeling feature extraction as a learned feature evolution process inspired by reaction–diffusion dynamics. More broadly, physics-informed neural networks [24] and Fourier-based operators [25] have demonstrated the value of embedding physical priors into learning systems, motivating the integration of such mechanisms into high-level vision tasks.

However, the proposed detector is not intended as a physics-informed PDE solver. Unlike PINNs, which incorporate governing equations as explicit training constraints, and unlike Fourier-based neural operators, which parameterize mappings in the spectral domain for PDE-related problems, RD-DETR uses reaction–diffusion-inspired structure as an inductive bias for learned feature evolution in object detection.

3. RD-DETR Architecture

3.1. Overall Framework

RD-DETR extends RT-DETR with three components (Figure 1): the RDNet backbone, a Phase-Guided Hybrid Encoder, and a Transformer decoder. RDNet extracts multi-scale features $\{S_3, S_4, S_5\}$, where shallow layers preserve fine-grained structural details through standard feature extraction, and deep layers apply reaction–diffusion dynamics to suppress accumulated noise and enhance target representations. Within the encoder, PGSA processes shallow features S_3 and S_4 to recover frequency domain structural details, while AIFI facilitates semantic interactions in the deep feature S_5. For cross-scale feature fusion, CA-AFM replaces static concatenation with dynamic content-aware aggregation. The resulting multi-scale features are passed to the Transformer decoder, which generates final detections through IoU-aware query selection. The three components address noise modeling, structural attention, and adaptive fusion in a coordinated manner, forming a coherent pipeline for robust vehicle detection under degraded conditions.

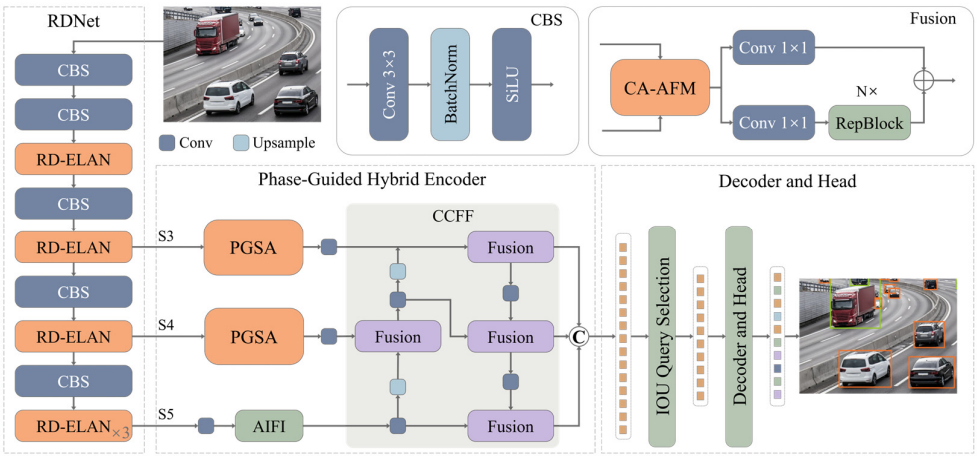


Figure 1. RD-DETR architecture overview.

3.2. Physics-Inspired Reaction–Diffusion Backbone

Traditional CNN backbones suffer feature degradation under challenging imaging conditions, as fixed convolution kernels cannot adaptively handle diverse degradation patterns such as atmospheric scattering, rain/snow noise, and sensor artifacts. To address this, we propose RDNet, which incorporates reaction–diffusion-inspired dynamics into feature extraction.

This design is conceptually related to TNRD [12], which learns reaction–diffusion parameters for low-level image restoration such as denoising and super-resolution. RDNet

differs from such approaches in both task and training objective: the reaction–diffusion dynamics are applied at the feature level inside a detection backbone, and the corresponding update operators are jointly optimized with the downstream detection objective rather than a pixel-reconstruction loss. The role of reaction–diffusion thus shifts from an explicit image-restoration solver to a structural inductive bias for feature learning under adverse imaging conditions.

3.2.1. Reaction–Diffusion Module

Reaction–diffusion systems [26] form stable spatiotemporal patterns through coupled diffusion and reaction processes, making them naturally suited to feature extraction: diffusion suppresses noise while reaction amplifies discriminative features. In a continuous spatiotemporal domain, the feature state $\boldsymbol{U}(x, y, t)$ evolves according to the partial differential equation:

$$\frac{\partial \boldsymbol{U}}{\partial t} = \nabla \cdot (\boldsymbol{D} \nabla \boldsymbol{U}) + \mathscr{R}(\boldsymbol{U}) \tag{1}$$

where $\boldsymbol{D}$ is the diffusion coefficient matrix and $\mathscr{R}(\cdot)$ is a nonlinear reaction functional.

For notational brevity, we denote the diffusion operator in Equation (1) as $\mathscr{D}(\cdot) := \nabla \cdot (\boldsymbol{D} \nabla \cdot)$. The learnable discrete diffusion tensor used in RD-Block is denoted by $\boldsymbol{D}_d$ to distinguish it from the continuous coefficient matrix $\boldsymbol{D}$. Starting from this continuous formulation, we make two modifications to obtain a network-friendly form. First, we augment the reaction term with an interaction term that couples the evolving state with the initial spatial features. Second, we apply an explicit Euler discretization with time step Δt, yielding a residual update of the form:

$$\boldsymbol{U}^{k+1} = \boldsymbol{U}^{k} + \Delta t [\mathscr{D}(\boldsymbol{U}^{k}) + \mathscr{R}(\boldsymbol{U}^{k}) + \mathscr{I}(\boldsymbol{U}^{k}, \boldsymbol{S})] \tag{2}$$

where $\mathscr{D}(\cdot)$, $\mathscr{R}(\cdot)$, and $\mathscr{I}(\cdot, \cdot)$ denote the diffusion, reaction, and interaction operators, respectively. Equation (2) provides a conceptual Euler-style transition from the continuous formulation, whereas Equation (3) gives the implemented hidden-state update in RD-Block. Based on this interpretation, we design a Reaction–Diffusion Block (RD-Block) that implements the above learned dynamical update in the network. As shown in Figure 2a, given input features $\boldsymbol{X} \in \mathbb{R}^{C \times H \times W}$, the RD-Block first projects them to a hidden state $\boldsymbol{u_0}$ via 1×1 convolution, followed by layer normalization, and separately extracts spatial features $\boldsymbol{s}$ via 3×3 depthwise convolution for subsequent state interaction. The state then evolves over T discrete steps; the k-th update is given by:

$$\boldsymbol{u}_{k+1} = \mathrm{GELU}(\boldsymbol{u}_k + \mathscr{F}_{\mathrm{diff}}(\boldsymbol{u}_k) + \mathscr{F}_{\mathrm{react}}(\boldsymbol{u}_k) + \mathscr{F}_{\mathrm{int}}(\boldsymbol{u}_k, \boldsymbol{s})) \tag{3}$$

where $\mathscr{F}_{\mathrm{diff}}$, $\mathscr{F}_{\mathrm{react}}$, and $\mathscr{F}_{\mathrm{int}}$ denote the diffusion, reaction, and interaction terms, respectively. GELU is used here as a smooth nonlinear surrogate that improves optimization stability while preserving nonlinear response capacity; it is not intended to represent an exact physical reaction law. In the implemented RD-Block, the effect of the time-step scaling is absorbed into the learnable operators. Therefore, Equation (3) should be viewed as the parameterized neural realization of the conceptual update in Equation (2), rather than as an exact numerical solver of a physical PDE. Accordingly, the iteration number T is treated as a structural hyperparameter of RD-Block, and its influence on detection accuracy and computational cost is examined in Section 4.3.3.

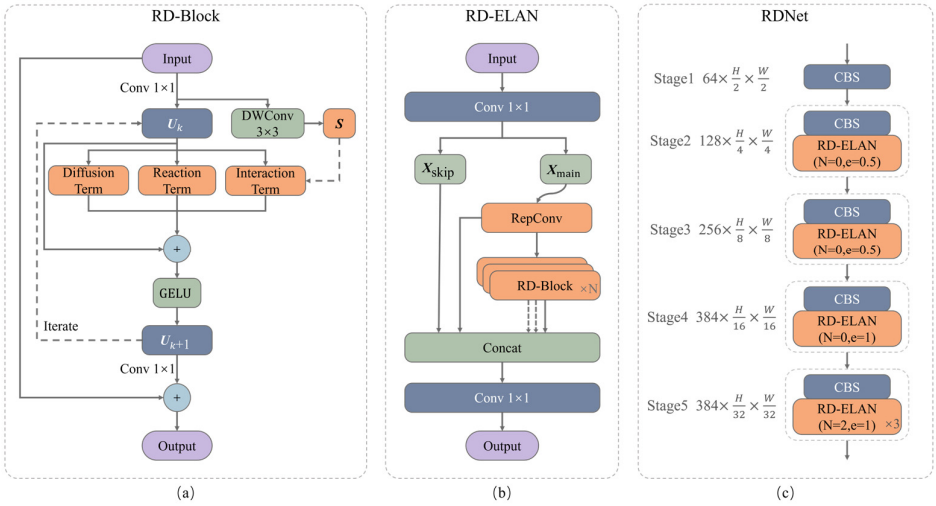


Figure 2. RDNet backbone structure. (**a**) RD-Block; (**b**) RD-ELAN module; (**c**) RDNet backbone architecture.

The diffusion term implements learnable adaptive diffusion:

$$\mathcal{F}_{\text{diff}}(\boldsymbol{u}) = \boldsymbol{D}_d \odot \nabla^2 \boldsymbol{u} \tag{4}$$

where $\boldsymbol{D}_d$ denotes the learnable discrete diffusion coefficient tensor associated with the continuous coefficient matrix $\boldsymbol{D}$ introduced in Equation (1), $\odot$ denotes element-wise multiplication, and ∇^2 is the discrete Laplacian operator realized via a learnable 3×3 convolution kernel. Accordingly, the diffusion in RD-Block is therefore not controlled by a fixed scalar hyperparameter, but by the learnable coefficient tensor $\boldsymbol{D}_d$ in Equation (4), which is optimized jointly with the rest of the detector.

The reaction term applies a nonlinear transformation through two successive 1×1 convolutions:

$$\mathcal{F}_{\text{react}}(\boldsymbol{u}) = \text{Conv}_{1\times1}(\text{GELU}(\text{Conv}_{1\times1}(\boldsymbol{u}))) \tag{5}$$

The interaction term fuses the current state with the initial spatial features:

$$\mathcal{F}_{\text{int}}(\boldsymbol{u}, \boldsymbol{s}) = \text{Conv}_{1\times1}(\text{GELU}(\text{Conv}_{1\times1}(\text{Concat}[\boldsymbol{u}, \boldsymbol{s}]))) \tag{6}$$

After T iterative evolution steps, an output projection maps the final hidden state back to the target channel dimension.

3.2.2. RD-ELAN Module

To balance computational efficiency with degradation-aware feature learning, we propose the RD-ELAN module, inspired by the Efficient Layer Aggregation Network (ELAN) [4]. As shown in Figure 2b, the module adopts a split–evolve–aggregate design that decouples feature evolution from the gradient flow path. Input features are first split via a 1×1 convolution into a gradient-preserving path $\boldsymbol{X}_{\text{skip}}$ and a feature-evolution path $\boldsymbol{X}_{\text{main}}$. Following initial feature extraction by RepConv [27], $\boldsymbol{X}_{\text{main}}$ passes through a sequence of N cascaded RD-Blocks. RepConv employs multi-branch structures during training and fuses them into a single 3×3 convolution via structural reparameterization at inference, reducing latency without sacrificing representational capacity.

The module supports flexible configuration via two hyperparameters: the evolution depth N controls the number of RD-Blocks, and the channel expansion ratio e controls intermediate channel dimensions. When $N = 0$, the module operates in fast mode with only RepConv and a transition convolution; when $N > 0$, features undergo multi-step reaction–

diffusion evolution. Finally, the module densely aggregates $\boldsymbol{X}_{\text{skip}}$ with all intermediate evolution states:

$$\boldsymbol{Y} = \text{Conv}_{1\times1}(\text{Concat}[\boldsymbol{X}_{\text{skip}}, \boldsymbol{X}_{\text{main}}^{(0)}, \boldsymbol{X}_{\text{main}}^{(1)}, \ldots, \boldsymbol{X}_{\text{main}}^{(N)}]) \tag{7}$$

where $\boldsymbol{X}_{\text{main}}^{(0)}$ is the RepConv output and $\boldsymbol{X}_{\text{main}}^{(i)}$ is the output of the i-th RD-Block.

3.2.3. RDNet Architecture

Built on RD-ELAN modules, RDNet adopts a pyramid-based enhancement strategy with differentiated configurations across stages, reflecting the distinct roles of features at different depths. Shallow features are rich in texture and spatial resolution but sensitive to over-smoothing; deep features are highly abstract but disproportionately vulnerable to noise accumulation. As shown in Figure 2c, RDNet configures each stage accordingly. Stage 1 applies standard convolution for efficient downsampling and basic texture extraction. From Stage 2 onward, RD-ELAN modules are deployed: Stages 2–4 set $N = 0$, with $e = 0.5$ for Stages 2–3 (lightweight) and $e = 1.0$ for Stage 4, where RepConv preserves high-frequency details such as vehicle edges. Stage 5 sets $N = 2$ and $e = 1.0$, with three stacked RD-ELAN modules; the reaction–diffusion evolution at this depth suppresses accumulated noise while enhancing target-level representations. RDNet outputs multi-scale features $\{\boldsymbol{S}_3, \boldsymbol{S}_4, \boldsymbol{S}_5\}$ at 8×, 16×, and 32× downsampling for subsequent multi-scale fusion.

3.3. Phase-Guided Hybrid Encoder

Multi-scale features extracted by RDNet must be enhanced and fused prior to detection. Conventional fusion strategies, including channel-wise concatenation and element-wise addition, operate entirely in the spatial domain and rely implicitly on magnitude-based representations, which are susceptible to degradation under adverse imaging conditions such as low illumination and heavy precipitation. To overcome this limitation, we design a Phase-Guided Hybrid Encoder consisting of two components: a Phase-Guided Spatial Attention module (PGSA) and a Content-Aware Adaptive Fusion Module (CA-AFM).

3.3.1. Phase-Guided Spatial Attention (PGSA)

The phase spectrum of an image is known to encode structural information, such as edges and contours, and to be relatively less sensitive to global illumination and contrast variations than the magnitude spectrum [14]. Motivated by this property, PGSA generalizes phase-based structural cues from the image domain to the feature map domain, using frequency domain phase content as an inductive bias for spatial attention. The empirical effectiveness of this generalization, which is not directly implied by image domain results, is evaluated in Section 4.3.4.

PGSA also differs in design intent from existing frequency domain methods. Unlike GFNet [13], which applies learnable global filters in the frequency domain, and unlike Fourier-based neural operators [25], which parameterize mappings in the spectral domain, PGSA explicitly uses phase-derived features to drive spatial attention. The proposed design therefore complements magnitude-based or full-spectrum frequency modeling rather than replacing it.

Accordingly, the phase branch is interpreted as a learned feature-space structural prior that captures edge and contour cues, rather than as a strict physical model of phase in the image domain. As shown in Figure 3, given input features $\boldsymbol{X} \in \mathbb{R}^{C\times H\times W}$, PGSA first maps them to the frequency domain via the Fast Fourier Transform (FFT):

$$\boldsymbol{X}_{\text{fft}} = \mathscr{F}(\boldsymbol{X}) \tag{8}$$

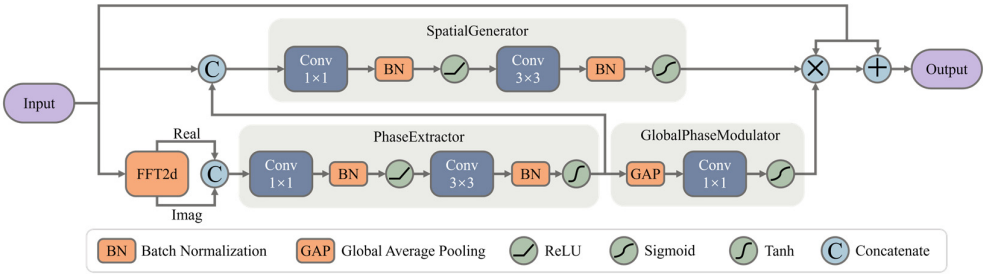


Figure 3. PGSA module structure.

Direct computation of the phase angle via arctangent suffers from numerical instability when the denominator approaches zero. Instead, the real and imaginary components are concatenated channel-wise and passed through a phase extractor Φ:

$$\boldsymbol{P} = \Phi(\text{Concat}[\text{Re}(\boldsymbol{X}_{\text{fft}}), \text{Im}(\boldsymbol{X}_{\text{fft}})]) \tag{9}$$

The joint real–imaginary representation retains both phase and magnitude information in a numerically stable form, while allowing the learnable extractor Φ to selectively emphasize phase-related structure; Section 4.3.4 validates this choice against magnitude-only and phase-only alternatives.

The phase extractor Φ consists of a 1×1 convolution followed by a 3×3 convolution, each succeeded by batch normalization; ReLU activation is applied between the two layers, and a tanh activation at the output bounds the phase representation to $[-1, 1]$. The output phase features $\boldsymbol{P} \in \mathbb{R}^{C' \times H \times W}$, where $C' = \max(8, \lfloor C/4 \rfloor)$, are kept compact to retain structural cues without incurring excessive computational overhead.

The phase features are then concatenated with the original spatial features and passed through a spatial attention generator Ψ to produce a channel-wise attention map:

$$\boldsymbol{A}_{\text{s}} = \Psi(\text{Concat}[\boldsymbol{X}, \boldsymbol{P}]) \tag{10}$$

The generator Ψ follows the same architectural pattern as Φ, with a Sigmoid activation at the output yielding $\boldsymbol{A}_{\text{s}} \in \mathbb{R}^{C \times H \times W}$.

To complement the local spatial modulation with a global context signal, a global phase modulator compresses the phase features via global average pooling and produces a channel-level gating vector:

$$\boldsymbol{G} = \sigma(\text{Conv}_{1\times1}(\text{GAP}(\boldsymbol{P}))) \tag{11}$$

where σ denotes the Sigmoid function and $\boldsymbol{G} \in \mathbb{R}^{C \times 1 \times 1}$. The final enhanced features are obtained by jointly applying spatial attention and global gating through a residual formulation:

$$\boldsymbol{Y} = \boldsymbol{X} \odot (1 + \boldsymbol{A}_{\text{s}} \odot \boldsymbol{G}) \tag{12}$$

The additive residual term $1 + \boldsymbol{A}_{\text{s}} \odot \boldsymbol{G}$ ensures that the original features are preserved when the attention response is weak, providing training stability in the early stages of optimization.

3.3.2. Content-Aware Adaptive Fusion Module (CA-AFM)

Conventional multi-scale fusion methods rely on fixed-weight channel concatenation or element-wise addition, which are agnostic to input content and therefore unable to adapt to varying scene conditions. We propose the Content-Aware Adaptive Fusion Module (CA-AFM), which achieves dynamic feature aggregation by learning content-dependent fusion weights.

As shown in Figure 4, given multi-scale input features $\{\boldsymbol{X}_i\}$ $(i = 1, \ldots, n)$, CA-AFM first projects all features to a unified channel dimension d via 1×1 convolution. A mean feature map is then computed as a compact scene-level summary:

$$\overline{\boldsymbol{F}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_i \tag{13}$$

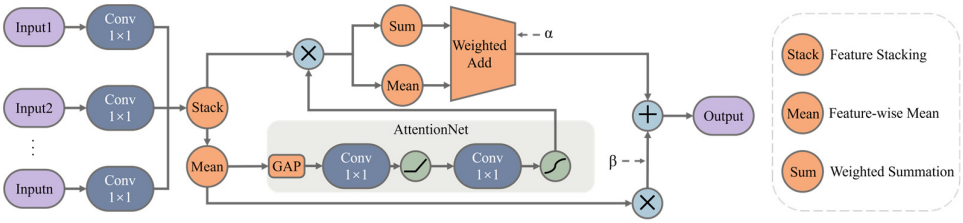


Figure 4. CA-AFM module structure. The scalar parameters α_ℓ and β_ℓ are learned independently at each CA-AFM position in the network; their per-position learned values are reported in Section 4.3.5.

A lightweight attention network estimates the importance weight of each scale based on this summary:

$$\boldsymbol{w} = \sigma(\mathrm{Conv}(\mathrm{ReLU}(\mathrm{Conv}(\mathrm{GAP}(\overline{\boldsymbol{F}}))))) \tag{14}$$

where $\boldsymbol{w} = [w_1, w_2, \ldots, w_n]^\top \in \mathbb{R}^n$ is the vector of per-scale importance weights and σ denotes the Sigmoid function.

The fusion at the ℓ-th CA-AFM position is controlled by two learnable scalar parameters: α_ℓ and β_ℓ. Since $\boldsymbol{Y}_{\text{mean}}$ in Equation (15) is a mean-scaled version of $\boldsymbol{Y}_{\text{sum}}$, α_ℓ modulates the effective magnitude of the content-aware weighted aggregation rather than interpolating between two independent fusion streams. The parameter β_ℓ is a learnable scalar gain on the unweighted mean-feature term $\overline{\boldsymbol{F}}$, allowing the model to reinforce or suppress the shared mean-feature response at each CA-AFM position. Both parameters are learned independently at each CA-AFM position and optimized jointly with the downstream detection loss. Their learned values are reported in Section 4.3.5. The weighted components are computed as:

$$\boldsymbol{Y}_{\text{sum}} = \sum_{i=1}^{n} \boldsymbol{w}_i \odot \boldsymbol{X}_i, \quad \boldsymbol{Y}_{\text{mean}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{w}_i \odot \boldsymbol{X}_i \tag{15}$$

and the final fused output is:

$$\boldsymbol{Y} = \alpha_\ell \boldsymbol{Y}_{\text{sum}} + (1 - \alpha_\ell)\boldsymbol{Y}_{\text{mean}} + \beta_\ell \overline{\boldsymbol{F}} \tag{16}$$

This parameterized formulation enables CA-AFM to adapt its fusion behavior at each position: a larger α_ℓ increases the effective contribution of the unnormalized weighted aggregation, helping preserve fine-grained, scale-specific structural signals, whereas a smaller α_ℓ attenuates this contribution. The $\beta_\ell \overline{\boldsymbol{F}}$ term independently modulates the unweighted mean-feature contribution; positive β_ℓ reinforces this term, while negative β_ℓ can offset redundant mean-feature responses already captured by the weighted aggregation. In implementation, α_ℓ and β_ℓ are initialized to 0.7 and 0.1, respectively, reflecting a moderate initial weighting of the content-aware weighted aggregation together with a small positive mean-feature gain; these initialization values are used across all experiments reported in this paper.

4. Experiments

We evaluate RD-DETR on two benchmarks, BDD100K and DAWN, through comparisons with representative detectors, ablation studies, and visualization analysis, with particular attention to detection accuracy under adverse imaging conditions and computational efficiency.

4.1. Experimental Setup

4.1.1. Datasets

BDD100K [28] is a large-scale driving dataset covering diverse weather conditions, illumination levels, and scene types. In this study, we use its official labeled validation subset of 10,000 images and divide it into 7000 training, 2000 validation, and 1000 test images. The partition follows a class-balance-oriented principle so that the per-class proportions remain close across the three subsets, as reflected in the instance counts reported in Table 1. The resulting split is fixed: the exact filename lists of the training, validation, and test subsets are provided in the Supplementary Materials, and the same split is reused for all baseline detectors, ablation variants, and repeated-run experiments to ensure that every method is compared under identical data and evaluation conditions.

Table 1. Per-class instance counts of the adopted train/validation/test splits.

Dataset	Split	Images	Bicycle *	Bus	Car	Motorcycle *	Person	Rider	Truck	Total
BDD100K	Train	7000	739	1143	71,658	310	9358	468	3005	86,681
BDD100K	Validation	2000	214	338	20,881	94	2676	132	862	25,197
BDD100K	Test	1000	79	179	10,370	56	1388	57	377	12,506
DAWN	Train	700	20	120	4523	61	330	N/A	457	5511
DAWN	Validation	200	5	26	1211	7	94	N/A	123	1466
DAWN	Test	100	1	16	607	13	44	N/A	61	742

* The bicycle and motorcycle columns refer to BDD100K's "bike" and "motor" classes and DAWN's "bicycle" and "motorcycle" classes, respectively. N/A indicates that the corresponding category is not annotated in the dataset.

To examine the representativeness of the adopted image set, Figure 5 compares its distribution with the complete labeled BDD100K collection of 80,000 images in terms of weather, scene, time of day, and object class. The largest per-bin share difference is at most 1 percentage point, indicating that the adopted image set preserves the main environmental and semantic characteristics of the larger labeled collection. Table 1 further reports the per-class instance counts of the training, validation, and test subsets, and the class proportions remain close across the three splits.

For DAWN [29], the 1000 annotated images are divided into 700 training, 200 validation, and 100 test images following the same class-balance-oriented principle. As DAWN is smaller and more imbalanced than BDD100K, the per-class instance counts of each split are reported in Table 1 to make the evaluation setting transparent. The exact filename lists of the three subsets are likewise provided in the Supplementary Materials.

This study adopts a multi-class detection setting using the original annotation categories of each dataset. For BDD100K, the evaluated categories are bike, bus, car, motor, person, rider, and truck; for DAWN, the evaluated categories are bicycle, bus, car, motorcycle, person, and truck. The detailed filename lists of the adopted train, validation, and test subsets are provided in the Supplementary Materials.

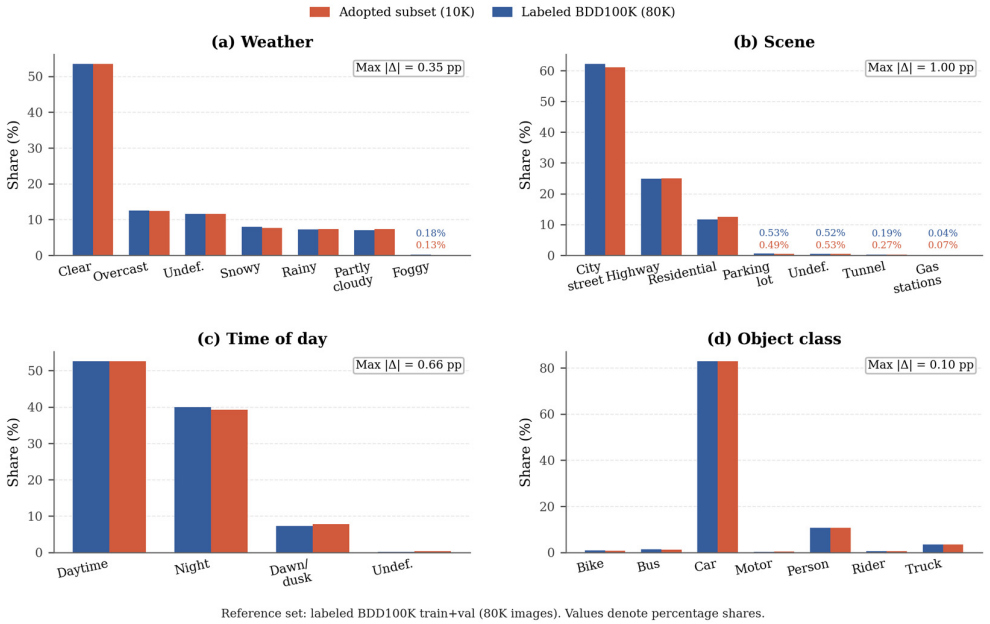


Figure 5. Distribution comparison of the adopted subset and labeled BDD100K.

4.1.2. Implementation Details

All experiments are conducted using PyTorch 2.2.1 with CUDA 11.8, with RT-DETR-R18 as the baseline. For RT-DETR and RD-DETR, models are trained for 300 epochs at an input resolution of 640 × 640, with a batch size of four. Optimization follows the AdamW optimizer with an initial learning rate of 1×10^{-4} and a cosine annealing schedule. Data augmentation includes Mosaic and MixUp strategies. The best checkpoint of each model is selected on the validation set under the same evaluation protocol.

All baseline results reported in this study are reproduced by the authors rather than taken directly from the original papers. MMDetection is used for conventional and DETR-family baselines, Ultralytics for RT-DETR and Ultralytics-based YOLO baselines, and the official YOLOv13 implementation for YOLOv13l. All compared models are trained and evaluated on the same data split, task definition, and evaluation code. For detectors with substantially different optimization characteristics, we follow the recommended training recipes of the corresponding implementations rather than forcing a single unified hyperparameter setting across architectures. The source implementation and key training settings of each reproduced baseline are summarized in Table 2. For RT-DETR and RD-DETR, the main comparison in Table 3 reports the run under seed 42, while repeated-run results over three training seeds (41, 42, and 43) are additionally reported in Table 4 to assess run-to-run stability.

Table 2. Source codebases and training settings used for baseline reproduction.

Model	Codebase	Epochs	Optimizer	LR	Batch
Faster R-CNN, Cascade R-CNN	MMDetection	12 (1×)	SGD	2×10^{-2}	16
RetinaNet, TOOD	MMDetection	12 (1×)	SGD	1×10^{-2}	16
DETR	MMDetection	150	AdamW	1×10^{-4}	16
DDQ-DETR	MMDetection	12	AdamW	2×10^{-4}	16
DAB-DETR	MMDetection	50	AdamW	1×10^{-4}	16
YOLOv9m, YOLOv10m, YOLO11m, YOLOv12m	Ultralytics	300	SGD/AdamW	$1 \times 10^{-2}/1 \times 10^{-3}$	16
YOLO26m	Ultralytics	300	MuSGD/AdamW	$1 \times 10^{-2}/1 \times 10^{-3}$	16

Table 2. *Cont.*

Model	Codebase	Epochs	Optimizer	LR	Batch
YOLOv13l	YOLOv13 (official)	300	SGD/AdamW	$1 \times 10^{-2}/1 \times 10^{-3}$	16
RT-DETR	Ultralytics	300	AdamW	1×10^{-4}	4
RD-DETR	This work	300	AdamW	1×10^{-4}	4

Note: Values separated by "/" refer to BDD100K and DAWN, respectively. For Ultralytics-based models, the optimizer is automatically selected by the framework based on the total number of training iterations. For fair comparison, all YOLO variants and YOLOv13l adopt a unified 300-epoch, batch-size-16 setting rather than each model's official large-scale COCO configuration. MMDetection-based models use the default input resolution of 768 × 1344, while all other models use 640 × 640. Batch size is reported per GPU; for RT-DETR and RD-DETR, this value matches the per-GPU batch size used in the official RT-DETR configuration.

Table 3. Comparison of detection performance on the adopted BDD100K test split.

Model	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	GFLOPs	Parameters (M)
CNN-based models						
Faster R-CNN	49.3	43.9	45.2	24.9	208.0	41.4
RetinaNet	60.6	43.3	46.5	25.2	208.2	36.5
TOOD	57.5	45.0	47.0	26.2	199.0	32.0
Cascade R-CNN	48.5	45.5	46.9	25.6	235.8	69.2
YOLO26m	64.8	44.0	47.7	27.2	67.9	20.4
YOLOv13l	59.8	47.1	48.4	27.5	88.1	27.5
YOLOv12m	63.4	44.8	47.9	27.3	59.5	19.6
YOLO11m	65.2	43.1	47.4	27.0	67.7	20.0
YOLOv10m	64.5	42.8	46.3	26.5	58.9	15.3
YOLOv9m	68.5	42.4	47.5	27.1	76.5	20.0
DETR-based models						
DETR	65.0	46.1	47.3	23.9	96.5	41.6
DDQ-DETR *	61.5	44.2	47.1	26.0	N/A	N/A
DAB-DETR	63.5	43.6	47.2	26.3	102.2	47.7
RT-DETR	64.3	45.1	48.2	27.7	57.0	19.9
RD-DETR	67.2	47.7	51.4	30.0	47.6	14.4

* GFLOPs and parameters for DDQ-DETR are not reported because comparable complexity statistics were not available for the reproduced configuration. DDQ-DETR is therefore included for accuracy comparison only. The RT-DETR and RD-DETR entries correspond to the main comparison run under seed 42; repeated-run results over training seeds 41, 42, and 43 are reported in Table 4.

Table 4. Repeated-run statistics of RT-DETR and RD-DETR on BDD100K (3 seeds).

Model/Metric	Seed 41	Seed 42	Seed 43	Mean ± Std	95% CI
RT-DETR mAP@0.5	47.9	48.2	47.8	48.0 ± 0.2	[47.45, 48.48]
RD-DETR mAP@0.5	51.2	51.4	51.1	51.2 ± 0.2	[50.85, 51.61]
RT-DETR mAP@0.5:0.95	27.3	27.7	27.8	27.6 ± 0.3	[26.94, 28.26]
RD-DETR mAP@0.5:0.95	29.6	30.0	29.8	29.8 ± 0.2	[29.30, 30.30]

4.1.3. Evaluation Metrics

We adopt standard object detection metrics. Precision, recall, Average Precision (AP), and mean Average Precision (mAP) are defined as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{17}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{18}$$

$$\text{AP} = \int_0^1 \text{Precision}(\text{Recall}) d(\text{Recall}) \tag{19}$$

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{20}$$

where TP, FP, and FN denote true positive, false positive, and false negative, respectively, and N is the number of object categories. The primary metric is mAP@0.5, which reflects overall detection capability; mAP@0.5:0.95 is additionally reported to evaluate high-precision localization. Model complexity is assessed via parameter count (Params) and floating-point operations (GFLOPs), which together characterize the computational cost of each method.

4.2. Comparative Experiments

4.2.1. Comparison on BDD100K Dataset

To comprehensively evaluate RD-DETR, we compare it against a range of mainstream detectors, including two-stage detectors (Faster R-CNN [1] and Cascade R-CNN [2]), single-stage detectors (RetinaNet [3], TOOD [30], and the YOLO series [5–8]), and Transformer-based detectors (DETR [9], DDQ-DETR [31], DAB-DETR [32], and RT-DETR [10]). Quantitative results are presented in Table 3. Figure 6 further illustrates the accuracy–efficiency trade-off across all evaluated models.

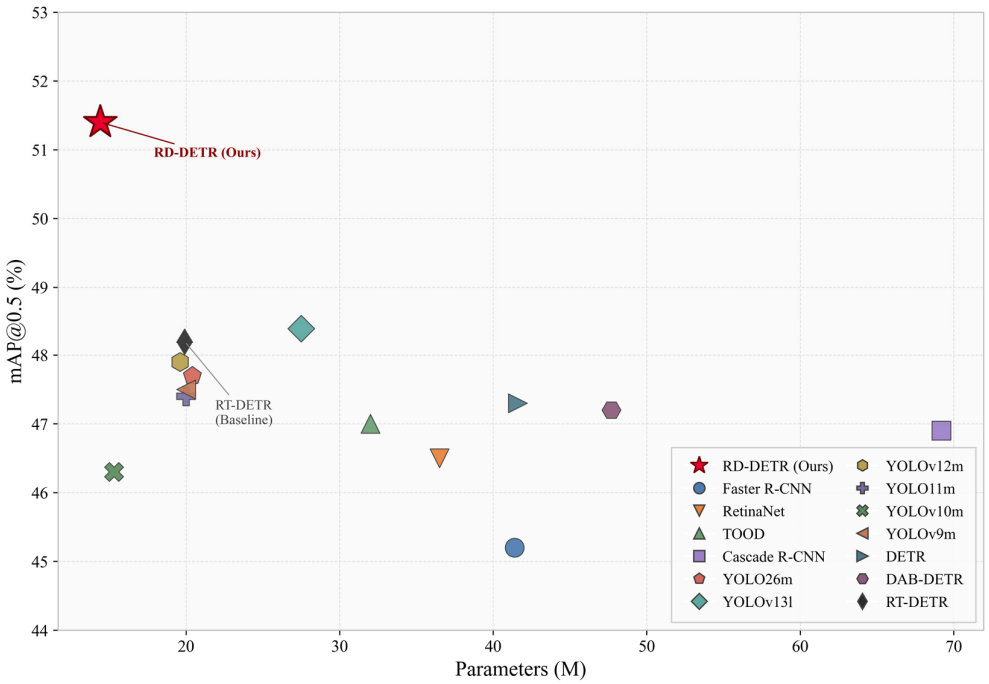


Figure 6. Accuracy–efficiency trade-off on BDD100K.

Comparison with the baseline. Compared to RT-DETR, RD-DETR improves mAP@0.5 by 3.2 percentage points and mAP@0.5:0.95 by 2.3 percentage points while simultaneously reducing parameters by 27.6% (14.4 M vs. 19.9 M) and GFLOPs by 16.5% (47.6 G vs. 57.0 G). The concurrent gains in accuracy and efficiency suggest that the proposed RDNet backbone provides a favorable representation–efficiency trade-off within the evaluated setting. To assess run-to-run stability, we trained RT-DETR and RD-DETR with three independent random seeds on BDD100K while keeping all other settings identical. As summarized in Table 4, RD-DETR achieves a higher mean mAP@0.5 than RT-DETR with limited variation across the tested seeds, suggesting that the observed gain is not attributable to a single favorable run.

Comparison with two-stage detectors. RD-DETR substantially outperforms both two-stage baselines in accuracy while being considerably more lightweight. Against Cascade

R-CNN, RD-DETR achieves a 4.5 percentage point gain in mAP@0.5 with only 20.8% of its parameters (14.4 M vs. 69.2 M) and roughly one-fifth of its GFLOPs (47.6 G vs. 235.8 G). The advantage over Faster R-CNN is similarly pronounced: a 6.2 percentage point improvement in mAP@0.5 alongside a 65.2% reduction in parameters.

Comparison with single-stage detectors. RD-DETR consistently surpasses all YOLO variants in detection accuracy. Among models with comparable parameter counts, namely, YOLOv12m (19.6 M), YOLO11m (20.0 M), and YOLO26m (20.4 M), RD-DETR leads by 3.5, 4.0, and 3.7 percentage points in mAP@0.5, respectively, while also consuming fewer GFLOPs. Against YOLOv13l, the strongest YOLO baseline in terms of mAP@0.5 (48.4%), RD-DETR leads by 3.0 percentage points with 47.6% fewer parameters. Notably, RD-DETR reports the lowest GFLOPs among the models in Table 3, with 47.6 G compared with 58.9 G for the lightweight YOLOv10m.

Comparison with Transformer-based detectors. Within the DETR family, RD-DETR achieves the highest mAP@0.5 (51.4%), surpassing DETR and DAB-DETR by 4.1 and 4.2 percentage points while requiring far fewer parameters (14.4 M vs. 41.6 M and 47.7 M, respectively). DDQ-DETR reaches a comparable mAP@0.5 of 47.1% but does not report parameter or GFLOPs figures, precluding a full efficiency comparison. Overall, RD-DETR provides the most favorable accuracy–efficiency trade-off among the compared Transformer-based baselines with available complexity statistics.

Using the matched training seeds, RD-DETR improves mAP@0.5 over RT-DETR by 3.27 ± 0.06 percentage points, with a 95% confidence interval of $[3.12,\ 3.41]$ percentage points. For mAP@0.5:0.95, the paired improvement is 2.20 ± 0.17 percentage points, with a 95% confidence interval of $[1.77,\ 2.63]$ percentage points. A two-sided paired t-test gives $p = 1.04 \times 10^{-4}$ for mAP@0.5 and $p = 2.06 \times 10^{-3}$ for mAP@0.5:0.95. Since only three training seeds are used, these results are interpreted as a repeated-run stability check for the primary BDD100K comparison rather than a comprehensive statistical study across all datasets.

4.2.2. Comparison on the DAWN Dataset

The DAWN dataset focuses on vehicle detection under adverse weather conditions, including fog, rain, snow, and sandstorms. Table 5 presents the comparative results, with RD-DETR achieving 59.9% mAP@0.5 and 36.8% mAP@0.5:0.95, surpassing all comparison models on both metrics.

Table 5. Comparison of detection performance on the adopted DAWN test split.

Model	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	GFLOPs	Parameters (M)
CNN-based models						
Faster R-CNN	77.3	39.7	47.7	26.1	208.0	41.4
RetinaNet	65.7	48.5	46.8	30.3	208.2	36.5
TOOD	66.6	49.9	49.5	29.8	199.0	32.0
Cascade R-CNN	74.2	47.6	49.7	31.4	235.8	69.2
YOLO26m	76.3	47.5	55.7	32.7	67.9	20.4
YOLOv13l	76.1	46.7	55.2	32.6	88.1	27.5
YOLOv12m	78.7	47.8	56.3	33.0	59.5	19.6
YOLO11m	76.8	47.7	56.0	33.0	67.7	20.0
YOLOv10m	71.3	49.3	53.6	32.8	58.9	15.3
YOLOv9m	75.6	49.5	55.1	34.2	76.5	20.0
DETR-based models						
DETR	71.8	48.9	52.0	28.6	96.5	41.6

Table 5. *Cont.*

Model	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	GFLOPs	Parameters (M)
DDQ-DETR *	68.3	46.2	50.8	29.4	N/A	N/A
DAB-DETR	70.1	49.6	53.9	29.9	102.2	47.7
RT-DETR	71.5	50.8	55.9	32.3	57.0	19.9
RD-DETR	74.5	54.2	59.9	36.8	47.6	14.4

* GFLOPs and parameters for DDQ-DETR are not reported because comparable complexity statistics were not available for the reproduced configuration. DDQ-DETR is therefore included for accuracy comparison only.

Compared to the baseline RT-DETR, RD-DETR improves mAP@0.5 by 4.0 percentage points (59.9% vs. 55.9%) and mAP@0.5:0.95 by 4.5 percentage points (36.8% vs. 32.3%). Against two-stage detectors, RD-DETR outperforms Cascade R-CNN by 10.2 percentage points in mAP@0.5 (59.9% vs. 49.7%) and Faster R-CNN by 12.2 percentage points (59.9% vs. 47.7%). Among single-stage detectors, YOLOv12m achieves the highest mAP@0.5 within the YOLO series (56.3%) yet still falls 3.6 percentage points below RD-DETR; the gap widens to 4.7 percentage points against YOLOv13l (55.2%) and 4.8 percentage points against YOLOv9m (55.1%), despite these models having substantially larger parameter counts. Within the DETR family, RD-DETR surpasses DETR and DAB-DETR by 7.9 and 6.0 percentage points in mAP@0.5, respectively. Notably, RD-DETR achieves larger gains over RT-DETR on DAWN than on BDD100K, indicating that the proposed design may offer stronger benefits as imaging conditions become more severe.

4.2.3. Weather-Specific Analysis

To further examine the robustness of RD-DETR under different degradation types, we report per-condition mAP@0.5 results on DAWN in Table 6. RD-DETR consistently outperforms RT-DETR across all four weather conditions, with improvements ranging from 2.9 to 4.8 percentage points.

Table 6. Per-condition detection performance on DAWN under adverse weather.

Weather Type	RT-DETR mAP@0.5 (%)	RD-DETR mAP@0.5 (%)	Improvement (pp)
Fog	56.1	59.8	+3.7
Rain	56.5	60.1	+3.6
Snow	54.4	59.2	+4.8
Sandstorm	60.4	63.3	+2.9
Overall	55.9	59.9	+4.0

The largest gain is observed under snow (+4.8 pp, 59.2% vs. 54.4%), where dense high-frequency disturbance poses a direct challenge to conventional feature extraction. This pattern is consistent with the intended role of reaction–diffusion-inspired feature evolution in suppressing structured degradation while preserving target-related responses. Under fog and rain, where low contrast and blur degrade spatial detail, RD-DETR improves mAP@0.5 by 3.7 and 3.6 percentage points, respectively. These gains are consistent with the intended role of phase-guided attention in preserving structural cues when spatial appearance becomes less reliable. Under sandstorm conditions, RT-DETR already achieves a relatively high baseline accuracy (60.4%), yet RD-DETR further improves to 63.3% (+2.9 pp), demonstrating consistent gains even when the baseline is strong. Overall, RD-DETR improves mAP@0.5 by 4.0 percentage points (59.9% vs. 55.9%) across all adverse weather conditions.

Taken together, these condition-specific results suggest that the reaction–diffusion and phase-guided components provide complementary benefits under different adverse imaging conditions: reaction–diffusion-inspired processing appears more beneficial under

structured noise, while phase-guided attention contributes useful structural cues when spatial appearance becomes less reliable.

4.3. Ablation Study

4.3.1. Overall Ablation

To validate the contribution of each proposed component, we conduct a layer-wise ablation study on BDD100K using RT-DETR-R18 as the baseline. The results are reported in Table 7.

Table 7. Ablation study on the adopted BDD100K test split.

Model	RDNet	PGSA	CA-AFM	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Parameters (M)	GFLOPs
Baseline				64.3	45.1	48.2	27.7	19.9	57.0
1	√			66.7	45.1	48.8	28.5	14.1	45.8
2		√		66.9	45.2	49.1	28.6	20.1	58.2
3			√	68.7	45.4	48.9	28.5	19.7	55.7
4	√	√		67.3	46.4	50.1	29.2	14.6	48.9
5	√		√	67.5	45.6	49.7	29.0	13.9	44.5
6		√	√	67.8	46.1	50.3	29.3	19.9	56.9
7	√	√	√	67.2	47.7	51.4	30.0	14.4	47.6

RDNet backbone. Replacing ResNet-18 with RDNet reduces parameters from 19.9 M to 14.1 M (−29.1%) and GFLOPs from 57.0 G to 45.8 G (−19.6%) while improving mAP@0.5 by 0.6 percentage points (48.2% → 48.8%). This result supports the effectiveness of the pyramid-based enhancement strategy: shallow stages operate in fast mode to preserve texture while reducing redundancy, whereas the deeper stage introduces reaction–diffusion-inspired evolution to suppress accumulated semantic noise.

PGSA module. Adding PGSA on top of RDNet further lifts mAP@0.5 to 50.1% (+1.3 pp) and recall from 45.1% to 46.4%. This improvement suggests that phase-guided cues help preserve structural information when spatial appearance is degraded, leading to more complete detections under the evaluated conditions.

CA-AFM module. Replacing static concatenation with CA-AFM on top of RDNet and PGSA achieves 51.4% mAP@0.5 (+1.3 pp) and 47.7% recall. CA-AFM employs learnable parameters α_ℓ and β_ℓ to modulate the effective strength of the content-aware weighted aggregation and the contribution of the unweighted mean-feature term $\overline{\boldsymbol{F}}$, helping reduce semantic mismatch that may arise from naive multi-scale aggregation.

Taken together, these results indicate that the three components provide complementary benefits across feature extraction, structural attention, and multi-scale aggregation, yielding a 3.2 percentage-point improvement in mAP@0.5 over the baseline (48.2% → 51.4%) and the largest gain among the evaluated component combinations.

4.3.2. Backbone Comparison

To evaluate the effectiveness of RDNet relative to alternative lightweight backbones, we replace ResNet-18 [33] with five representative architectures—StarNet [18], MobileNetV4 [19], FasterNet [17], LSKNet [34], and RDNet, —while keeping all other components fixed. The results are presented in Table 8.

Among the lightweight alternatives, FasterNet achieves the lowest parameter count (10.8 M) and GFLOPs (28.5 G), but at the cost of accuracy, reaching only 44.8% mAP@0.5, which is 3.4 percentage points below the ResNet-18 baseline (48.2%). StarNet and MobileNetV4 reach 46.1% and 45.6% mAP@0.5, respectively, both falling short of the baseline despite having substantially fewer parameters. LSKNet attains 46.7% mAP@0.5 through

large-kernel attention but exhibits notably low recall (42.6%), suggesting that its global receptive field comes at the expense of detection completeness.

Table 8. Comparison of detection performance across backbone architectures on BDD100K.

Model	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	GFLOPs	Parameters (M)
ResNet-18	64.3	45.1	48.2	27.7	57.0	19.9
StarNet	62.5	44.1	46.1	25.5	31.8	12.0
MobileNetV4	59.9	44.4	45.6	26.2	39.5	11.3
FasterNet	61.8	44.6	44.8	26.1	28.5	10.8
LSKNet	62.7	42.6	46.7	27.0	37.6	12.6
RDNet	66.7	45.1	48.8	28.5	45.8	14.1

RDNet achieves 48.8% mAP@0.5 and 28.5% mAP@0.5:0.95, surpassing all comparison backbones, as well as the ResNet-18 baseline (+0.6 pp), while reducing parameters by 29.1% (14.1 M vs. 19.9 M). Its GFLOPs (45.8 G) are higher than those of the other lightweight backbones, though still 19.6% below ResNet-18 (57.0 G). This reflects a deliberate design trade-off: RDNet accepts a moderate computational cost relative to purpose-built efficiency networks in exchange for stronger representation capability under the evaluated detection setting.

4.3.3. Parameter Sensitivity

We analyze the effect of the evolution step count T in RDNet on BDD100K. As shown in Table 9, increasing T from one to three improves mAP@0.5 from 48.3% to 51.4% (+3.1 pp), with GFLOPs increasing modestly from 46.4 G to 47.6 G. Further increases to $T = 4$ and $T = 5$ yield diminishing incremental gains of +0.2 pp each over the preceding step (+0.4 pp cumulative gain over $T = 3$), while GFLOPs continue to rise to 48.2 G and 48.8 G, respectively. Balancing accuracy and computational cost, we adopt $T = 3$ as the default configuration. This trend provides empirical guidance for choosing T in the detector-oriented feature-evolution setting.

Table 9. Sensitivity analysis of evolution step T on the adopted BDD100K test split.

T	mAP@0.5 (%)	mAP@0.5:0.95 (%)	GFLOPs
1	48.3	28.9	46.4
2	49.8	29.5	47.0
3	51.4	30.0	47.6
4	51.6	30.2	48.2
5	51.8	30.2	48.8

4.3.4. PGSA Frequency Component Ablation

To evaluate the contribution of different frequency domain components in PGSA, we compare four configurations on BDD100K that differ only in the input supplied to the phase extractor Φ (Equation (9)). The baseline removes PGSA entirely. The magnitude-only variant feeds Φ with the magnitude spectrum $|\boldsymbol{X}_{\text{fft}}|$, replicated along the channel dimension to match the default input shape. The phase-only variant feeds Φ with the normalized complex values $\boldsymbol{X}_{\text{fft}}/|\boldsymbol{X}_{\text{fft}}|$ (decomposed into real and imaginary parts), which retain phase-angle information on the unit circle while avoiding the 2π discontinuity of the raw phase angle. The default PGSA feeds Φ with the joint real–imaginary input $\text{Concat}[\text{Re}(\boldsymbol{X}_{\text{fft}}), \text{Im}(\boldsymbol{X}_{\text{fft}})]$, which retains both phase and magnitude information in a numerically stable form. PGSA applies FFT directly to each input feature map at its original spatial resolution, without introducing an additional Fourier resizing operation. Table 10

therefore focuses on the type of frequency information used by PGSA (magnitude, phase, or their joint representation) rather than on FFT resolution effects. The results are presented in Table 10.

Table 10. PGSA frequency component ablation on the adopted BDD100K test split.

Variant	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	GFLOPs	Params (M)
Baseline (no PGSA)	64.3	45.1	48.2	27.7	57.0	19.9
Magnitude only	64.9	45.0	48.5	27.9	58.2	20.1
Phase only	66.2	45.3	48.8	28.2	58.2	20.1
Phase + magnitude	66.9	45.2	49.1	28.6	58.2	20.1

The phase-only variant improves mAP@0.5 by 0.6 pp over the baseline, while the magnitude-only variant yields only 0.3 pp. The default PGSA, which retains both phase and magnitude information in a numerically stable form, achieves the largest gain (+0.9 pp over the baseline). These results suggest that phase-related structural cues contribute more strongly than magnitude-only cues in this setting, while the joint real–imaginary input provides the best overall performance by retaining complementary phase and magnitude information.

4.3.5. CA-AFM Fusion Strategy Ablation

To evaluate the effectiveness of CA-AFM, we compare five fusion strategies on BDD100K: (1) channel concatenation, (2) element-wise addition, (3) BiFPN-style weighted fusion, (4) CA-AFM with fixed $\alpha = 0.7, \beta = 0.1$, and (5) CA-AFM with learned α_ℓ and β_ℓ. The results are presented in Table 11.

Table 11. CA-AFM fusion strategy ablation on the adopted BDD100K test split.

Fusion Strategy	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	GFLOPs	Params (M)
Concat (baseline)	64.3	45.1	48.2	27.7	57.0	19.9
Element-wise Add	64.0	44.9	48.1	27.6	55.7	19.6
BiFPN-style	66.3	45.4	48.6	28.1	64.3	20.3
CA − AFM (fixed $\alpha = 0.7, \beta = 0.1$)	68.2	45.2	48.7	28.3	55.7	19.7
CA − AFM (learned α_ℓ and β_ℓ)	68.7	45.4	48.9	28.5	55.7	19.7

CA-AFM with learned parameters achieves the highest mAP@0.5 (48.9%), outperforming concatenation by 0.7 pp and BiFPN-style fusion by 0.3 pp. The content-aware weighted fusion accounts for most of this gain, and adaptive parameter learning provides a further 0.2 pp improvement over the fixed-parameter variant.

The converged values of α_ℓ and β_ℓ at the four CA-AFM positions, extracted from the trained RD-DETR, are summarized in Table 12. The four positions converge to distinctly different operating points: the deepest position (Layer 27) learns a weighted-aggregation-dominated setting with $\alpha_\ell = 0.888$ and $\beta_\ell = -0.034$, which is effectively close to zero; Layer 21 settles into the most balanced setting ($\alpha_\ell = 0.581$) with a strong positive $\beta_\ell = 0.201$, while Layer 24 leans moderately toward weighted aggregation ($\alpha_\ell = 0.701$) with a smaller positive $\beta_\ell = 0.126$; the shallowest position (Layer 15) converges to an $\alpha_\ell = 0.706$ nearly identical to that of Layer 24 but to a clearly negative $\beta_\ell = -0.106$, suggesting that the mean-feature term $\bar{F}$ can offset partial overlap with the weighted aggregation at this position. Training converged without divergence, with $\alpha_\ell \in [0.581, 0.888]$ and $\beta_\ell \in [-0.106, 0.201]$, reflecting the content-aware adaptive behavior of the design.

Table 12. Learned α_ℓ and β_ℓ at each CA-AFM position.

CA-AFM Position	α_ℓ	β_ℓ
Layer 15 (shallowest)	0.706	−0.106
Layer 21	0.581	0.201
Layer 24	0.701	0.126
Layer 27 (deepest)	0.888	−0.034

4.4. Visualization Analysis

Figure 7 illustrates representative detection results across multiple scenes. On the depicted scenes, RD-DETR produces more complete detections than the baseline under adverse imaging conditions, with fewer false positives and missed detections. Figure 8 compares feature maps extracted by different backbone architectures and shows that RDNet preserves clearer edge and boundary information in shallow layers while maintaining cleaner foreground–background separation in deeper layers, in line with the intended pyramid-based differentiated configuration strategy. Figure 9 shows feature activation heatmaps. Compared with the baseline, RD-DETR exhibits more concentrated responses over structurally informative regions, providing qualitative evidence consistent with the quantitative improvements. Figure 10 presents representative remaining failure cases of RD-DETR under degraded conditions. In low-visibility nighttime scenes, extremely small and distant vehicles may still be missed because their feature responses become too weak after repeated downsampling. In addition, visually similar vehicle categories may remain confused when degradation suppresses discriminative appearance cues, as illustrated by the misclassification of a bus as a truck in Figure 10a. These examples indicate that, despite the overall robustness gains, tiny-object detection and inter-class confusion among visually similar vehicle categories remain challenging under severe imaging degradation.



Figure 7. Detection comparison across representative scenes.

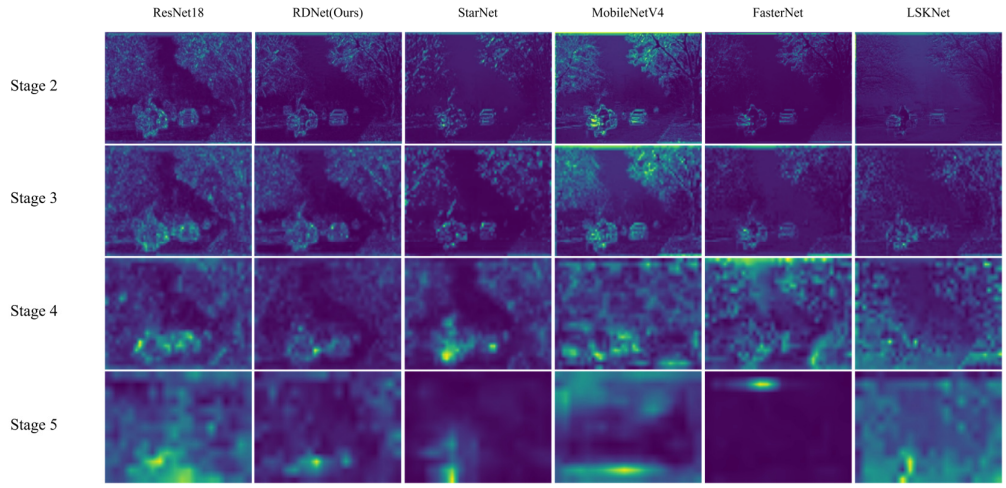


Figure 8. Feature map visualization across backbone architectures.



Figure 9. Feature activation heatmaps comparing RD-DETR and RT-DETR.

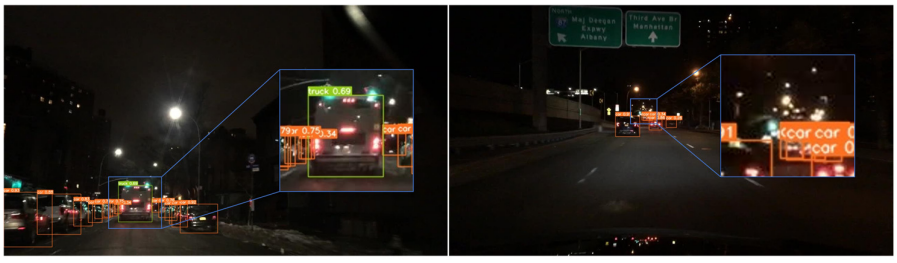


Figure 10. Representative remaining failure cases of RD-DETR under degraded conditions. (**a**) Misclassification of a bus as a truck in a nighttime scene. (**b**) Missed tiny distant vehicles in a low-visibility nighttime scene. Insets highlight the corresponding error regions.

5. Conclusions

This work addresses a fundamental tension in vehicle detection under degraded imaging conditions: existing deep detectors achieve strong benchmark performance yet lack explicit mechanisms to handle adverse imaging degradations commonly encountered in traffic scenes. Rather than treating adverse conditions merely as a data augmentation problem, we embed reaction–diffusion-inspired dynamics directly into the backbone and

model feature extraction as a learnable feature evolution process. Within the evaluated settings on BDD100K and DAWN, RD-DETR improves detection performance over RT-DETR while reducing model parameters and computational cost, supporting the usefulness of physics-inspired inductive biases for degradation-aware feature learning.

The goal of this study was to develop a detector that remains accurate and efficient under degraded imaging conditions. The reported evidence supports this goal within the evaluated settings. For RQ1, embedding reaction–diffusion-inspired dynamics into the backbone improves the detector's accuracy–efficiency trade-off. For RQ2, PGSA provides complementary structural cues under low-contrast and adverse weather conditions. For RQ3, content-aware adaptive fusion improves multi-scale aggregation relative to static fusion strategies. At the same time, the present work primarily addresses imaging degradations such as low illumination, adverse weather, blur, and sensor noise. Within degraded conditions, extreme small-object detection and fine-grained inter-class ambiguity remain open challenges; severe occlusion is a broader detector-wide difficulty orthogonal to imaging degradation that this work does not explicitly address. In addition, the BDD100K experiments are conducted on the adopted 10,000-image protocol so that all compared detectors, ablation variants, and repeated-run experiments can be evaluated under the same data split and evaluation pipeline; extending the evaluation to larger training protocols and additional adverse condition benchmarks is left as a direction for future work. The repeated-run analysis is also currently limited to the primary RT-DETR versus RD-DETR comparison on BDD100K, and extending multi-seed evaluation to DAWN and other benchmarks remains an important step toward a more comprehensive statistical assessment. Finally, the fusion parameters α_ℓ and β_ℓ in CA-AFM are parameterized without explicit bounds. While their learned values remained bounded throughout our experiments (Section 4.3.5), a bounded reparameterization—for example, parameterizing α_ℓ via a sigmoid and β_ℓ via a tanh function—could provide formal guarantees on parameter boundedness to complement the empirical evidence reported here and is a direction worth exploring in future work.

The three proposed components, namely, RDNet, PGSA, and CA-AFM, target distinct failure modes, including noise accumulation in deep semantic features, loss of structural detail under adverse imaging degradations, and semantic misalignment across scales during fusion. Their complementarity is reflected in the ablation results: each component yields consistent gains in isolation, and their combination yields the largest improvement among the evaluated component combinations. This modular design also suggests that individual components can be integrated into other detection architectures beyond RT-DETR.

Future work will explore several directions. Extending the physics-inspired design to handle compound degradation, where multiple adverse factors such as rain, fog, and sensor noise co-occur, represents a natural next step toward broader robustness in real traffic imaging conditions. On the architectural side, incorporating temporal information from consecutive video frames may further improve detection consistency and stability across time. More broadly, evaluating the framework on larger and more diverse benchmarks will help further assess the generalizability of physics-inspired inductive biases for degradation-aware perception.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/app16094378/s1: File S1: BDD100K train split; File S2: BDD100K validation split; File S3: BDD100K test split; File S4: DAWN train split; File S5: DAWN validation split; File S6: DAWN test split. Each file lists the image filenames used in the corresponding split, with one filename per line.

Author Contributions: Conceptualization, Y.H. and Y.C.; methodology, Y.H. and Y.C.; software, Y.H.; validation, Y.H., K.P. and H.H.; formal analysis, Y.H., K.P. and H.H.; investigation, Y.H., K.P., X.W. and H.H.; resources, X.W.; data curation, Y.H.; writing—original draft preparation, Y.H.; writing—review and editing, Y.C. and Y.M.; visualization, Y.H.; supervision, Y.C. and Y.M.; project administration, Y.M.; funding acquisition, Y.C. and Y.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Major Science and Technology Project of Guangxi, China (Grant No. AA23023011), and the Key Research and Development Program of Guangxi, China (Grant No. AB22080098).

Data Availability Statement: The BDD100K and DAWN datasets used in this study are publicly available from their respective official sources. The filename lists for the train/validation/test splits adopted in this study are provided in the Supplementary Materials. The implementation code of the proposed RDNet, PGSA, and CA-AFM modules, together with the key configuration files and training settings used in this study, is available from the corresponding author upon reasonable request.

Acknowledgments: The authors would like to thank Peng Wei for his assistance with data collection during this research.

Conflicts of Interest: Yishi Chen and Xiangning Wu are employees of Guangxi Research Institute of Mechanical Industry Co., Ltd. The remaining authors declare that this research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
2. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: Salt Lake City, UT, USA, 2018; pp. 6154–6162.
3. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*; IEEE: Venice, Italy, 2017; pp. 2999–3007.
4. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In *Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Vancouver, BC, Canada, 2023; pp. 7464–7475.
5. Wang, C.-Y.; Yeh, I.-H.; Liao, H.-Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616. [CrossRef]
6. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. In *Proceedings of the Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2024; Volume 37, pp. 107984–108011.
7. Tian, Y.; Ye, Q.; Doermann, D. YOLOv12: Attention-Centric Real-Time Object Detectors. *arXiv* **2025**, arXiv:2502.12524.
8. Lei, M.; Li, S.; Wu, Y.; Hu, H.; Zhou, Y.; Zheng, X.; Ding, G.; Du, S.; Wu, Z.; Gao, Y. YOLOv13: Real-Time Object Detection with Hypergraph-Enhanced Adaptive Visual Perception. *arXiv* **2025**, arXiv:2506.17733.
9. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In *Computer Vision—ECCV 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2020; Volume 12346, pp. 213–229.
10. Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; Chen, J. DETRs Beat YOLOs on Real-Time Object Detection. In *Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Seattle, WA, USA, 2024; pp. 16965–16974.
11. Tian, C.; Fei, L.; Zheng, W.; Xu, Y.; Zuo, W.; Lin, C.-W. Deep Learning on Image Denoising: An Overview. *Neural Netw.* **2020**, *131*, 251–275. [CrossRef] [PubMed]
12. Chen, Y.; Pock, T. Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1256–1272. [CrossRef] [PubMed]
13. Rao, Y.; Zhao, W.; Zhu, Z.; Zhou, J.; Lu, J. GFNet: Global Filter Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 10960–10973. [CrossRef] [PubMed]
14. Oppenheim, A.V.; Lim, J.S. The Importance of Phase in Signals. *Proc. IEEE* **1981**, *69*, 529–541. [CrossRef]

15. Lin, T.-Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Honolulu, HI, USA, 2017; pp. 936–944.
16. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Seattle, WA, USA, 2020; pp. 10778–10787.
17. Chen, J.; Kao, S.H.; He, H.; Zhuo, W.; Wen, S.; Lee, C.-H.; Chan, S.-H.G. Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks. In *Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Vancouver, BC, Canada, 2023; pp. 12021–12031.
18. Ma, X.; Dai, X.; Bai, Y.; Wang, Y.; Fu, Y. Rewrite the Stars. In *Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Seattle, WA, USA, 2024; pp. 5694–5703.
19. Qin, D.; Leichner, C.; Delakis, M.; Fornoni, M.; Luo, S.; Yang, F.; Wang, W.; Banbury, C.; Ye, C.; Akin, B.; et al. MobileNetV4: Universal Models for the Mobile Ecosystem. *arXiv* **2024**, arXiv:2404.10518.
20. Kang, L.; Lu, Z.; Meng, L.; Gao, Z. YOLO-FA: Type-1 Fuzzy Attention Based YOLO Detector for Vehicle Detection. *Expert Syst. Appl.* **2024**, *237*, 121209. [CrossRef]
21. Liu, J.; Yang, D.; Feng, T.; Fu, J. MDFD2-DETR: A Real-Time Complex Road Object Detection Model Based on Multi-Domain Feature Decomposition and De-Redundancy. *IEEE Trans. Intell. Veh.* **2025**, *10*, 4343–4359. [CrossRef]
22. Tahir, N.U.A.; Zhang, Z.; Asim, M.; Iftikhar, S.; Abd El-Latif, A.A. PVDM-YOLOv8l: A Solution for Reliable Pedestrian and Vehicle Detection in Autonomous Vehicles under Adverse Weather Conditions. *Multimed. Tools Appl.* **2025**, *84*, 27045–27070. [CrossRef]
23. Appiah, E.O.; Mensah, S. Object Detection in Adverse Weather Condition for Autonomous Vehicles. *Multimed. Tools Appl.* **2024**, *83*, 28235–28261. [CrossRef]
24. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations. *J. Comput. Phys.* **2019**, *378*, 686–707. [CrossRef]
25. Li, Z.; Kovachki, N.; Azizzadenesheli, K.; Liu, B.; Bhattacharya, K.; Stuart, A.; Anandkumar, A. Fourier Neural Operator for Parametric Partial Differential Equations. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 3–7 May 2021.
26. Turing, A.M. The Chemical Basis of Morphogenesis. *Philos. Trans. R. Soc. B Biol. Sci.* **1952**, *237*, 37–72. [CrossRef]
27. Ding, X.; Zhang, X.; Ma, N.; Han, J.; Ding, G.; Sun, J. RepVGG: Making VGG-Style ConvNets Great Again. In *Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Nashville, TN, USA, 2021; pp. 13728–13737.
28. Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; Darrell, T. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Seattle, WA, USA, 2020; pp. 2633–2642.
29. Kenk, M.A.; Hassaballah, M. DAWN: Vehicle Detection in Adverse Weather Nature Dataset. *arXiv* **2020**, arXiv:2008.05402. [CrossRef]
30. Feng, C.; Zhong, Y.; Gao, Y.; Scott, M.R.; Huang, W. TOOD: Task-Aligned One-Stage Object Detection. In *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: Montreal, QC, Canada, 2021; pp. 3490–3499.
31. Zhang, S.; Wang, X.; Wang, J.; Pang, J.; Lyu, C.; Zhang, W.; Luo, P.; Chen, K. Dense Distinct Query for End-to-End Object Detection. In *Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Vancouver, BC, Canada, 2023; pp. 7329–7338.
32. Liu, S.; Li, F.; Zhang, H.; Yang, X.; Qi, X.; Su, H.; Zhu, J.; Zhang, L. DAB-DETR: Dynamic Anchor Boxes Are Better Queries for DETR. In Proceedings of the International Conference on Learning Representations (ICLR), Virtual Event, 25–29 April 2022.
33. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Las Vegas, NV, USA, 2016; pp. 770–778.
34. Li, Y.; Hou, Q.; Zheng, Z.; Cheng, M.-M.; Yang, J.; Li, X. Large Selective Kernel Network for Remote Sensing Object Detection. In *Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: Paris, France, 2023; pp. 16748–16759.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.