

A Study on the Lightweighting of YOLOv11-based Traffic Sign Detection Models

Renke Chen *

School of Electronic Information and Artificial Intelligence, Shaanxi University of Science & Technology, Xi'an, Shaanxi, 710021, China

* Corresponding author Email: Anmixx@outlook.com

Abstract: Accurate and real-time traffic sign detection is a critical component of environmental perception for Advanced Driver-Assistance Systems (ADAS) and autonomous vehicles, playing a vital role in ensuring driving safety. However, current mainstream lightweight object detection models often encounter performance bottlenecks when dealing with challenges prevalent in real-world road scenes, such as small objects and complex backgrounds, primarily due to the loss of feature information. To address this issue, this paper conducts an in-depth application and performance evaluation of an improved YOLOv11-based lightweight detection algorithm. The algorithm is characterized by two key features: first, it enhances the perceptual capabilities for tiny objects by introducing a higher-resolution P2 feature layer; second, it adopts an integrated fusion detection module, termed FPN-P2 in this paper, which applies a "shallow feature direct-path, deep feature fusion" strategy to efficiently integrate multi-scale features. Extensive experiments on the public TT100K dataset demonstrate that the optimized configurations of this architecture achieve significant breakthroughs in both performance and efficiency. Specifically, its performance-optimal configuration achieves a mAP@0.5 of 76.3%, substantially surpassing baseline models like YOLOv8n. Meanwhile, its efficiency-optimal configuration attains a high accuracy of 68.8% while having fewer parameters than the baseline model. This research not only provides and validates highly competitive solutions for high-precision lightweight traffic sign detection but also, through intensive ablation studies, offers valuable empirical insights into model architecture selection and optimization for specific tasks.

Keywords: Traffic Sign Detection; YOLOv11; Small Object Detection; Feature Pyramid Network; Lightweight Model.

1. Introduction

1.1. Background and Significance

With the rapid development and profound application of artificial intelligence, Intelligent Transportation Systems (ITS) and autonomous driving have emerged as key domains for measuring a nation's technological innovation capabilities. Among these, Advanced Driver-Assistance Systems (ADAS), as a core technology for achieving autonomous driving, aim to assist drivers in operating vehicles safely through various onboard sensors and automated control technologies. According to a report by the World Health Organization (WHO), over one million people die in road traffic accidents annually, with human error being a primary cause[1]. By adopting a proactive prevention approach, ADAS can significantly reduce the incidence of such accidents, marking a revolutionary milestone in enhancing driving safety. Within the environmental perception module of ADAS, Traffic Sign Detection and Recognition (TSDR) stands out as a fundamental and crucial task. Traffic signs, serving as a standardized "road language," provide critical road information such as mandatory instructions, warnings, and guidance to both vehicles and drivers. Autonomous vehicles must interpret this information accurately and in real time to make safe decisions that comply with traffic regulations, thereby ensuring driving safety and traffic efficiency. Consequently, research into high-performance traffic sign detection algorithms is not only a technical necessity for advancing the real-world application of autonomous driving but also holds significant practical importance for improving overall road safety.

1.2. Literature Review

The development of object detection algorithms can be broadly divided into two phases. Early research primarily relied on traditional machine learning methods, such as the Viola-Jones detector using Haar-like features and approaches combining Histogram of Oriented Gradients (HOG) features with Support Vector Machines (SVMs)[2,3]. These methods depended on handcrafted feature extractors, which exhibited poor robustness in complex scenarios involving illumination changes and object occlusions, thus failing to meet the demands of practical applications.

In recent years, deep learning-based object detection algorithms, with Convolutional Neural Networks (CNNs) at their core, have achieved breakthrough progress and have become mainstream. Currently, these algorithms are mainly categorized into two types: two-stage detectors, represented by Faster R-CNN, and one-stage detectors, represented by the YOLO series[4]. Two-stage algorithms first generate candidate boxes via a Region Proposal Network (RPN) before performing classification and regression. While they offer high accuracy, their slower processing speed makes them unsuitable for the real-time requirements of autonomous driving.

One-stage algorithms, in contrast, frame detection as a regression problem, directly predicting bounding boxes and class labels to achieve an excellent balance between speed and accuracy. Among them, the YOLO (You Only Look Once) series has garnered significant attention for its outstanding performance. From the initial YOLO that unified the detection problem into a single regression task, to YOLOv3 which introduced multi-scale feature pyramids for prediction, and later YOLOv5 which achieved tremendous success in

model deployment and engineering practices, the YOLO series has established itself as a benchmark in the field of real-time object detection[5-7]. In recent years, the series has continued to iterate at an unprecedented pace: YOLOv8 further optimized performance by incorporating an Anchor-Free design and a C2f module; the subsequent YOLOv9 addressed the information bottleneck issue in deep networks through Programmable Gradient Information (PGI); and YOLOv10 achieved end-to-end detection via an NMS-Free design[8-10]. Most recently, explorations such as GOLD-YOLO have begun to reconstruct the network's feature fusion paradigm with novel mechanisms like "Gather-and-Distribute" to pursue higher efficiency and accuracy[11]. However, it is noteworthy that while these cutting-edge general-purpose model frameworks focus on universal architectural optimizations, they still do not provide a dedicated solution that tackles the performance bottlenecks caused by extremely small objects in specific scenes like traffic sign detection from the root level of detection scales.

1.3. Challenges

Despite the remarkable achievements of general-purpose object detection algorithms like YOLO, their application to the real-world task of traffic sign detection still faces several key challenges, especially when processing challenging datasets such as TT100K[12]. These challenges are outlined below:

(1) Severe Loss of Feature Information for Small Objects: According to the statistics of the TT100K dataset, a vast number of traffic signs are extremely small in the 2048×2048 high-resolution images[12]. Mainstream network models need to perform multiple downsampling operations during feature extraction to expand their receptive fields. However, this process can easily cause critical feature information of small objects, such as textures and contours, to be submerged or even completely lost, leading to severe missed detections.

(2) Strong Interference from Complex Backgrounds: The real-world road environment is complex and ever-changing. Drastic illumination changes like intense daylight and low light at night, adverse weather conditions such as rain and fog, and random occlusions from trees, buildings, and other vehicles can all severely impact a model's detection performance, posing stringent requirements on the algorithm's robustness.

(3) Accuracy Bottleneck of Lightweight Models: To meet the real-time deployment demands of edge devices (e.g., in-vehicle chips), lightweight network models (such as YOLOv5n and YOLOv8n) have become a research hotspot. However, in their pursuit of low parameter counts and computational costs, these models typically simplify their network architecture and reduce the number of channels, which further weakens their feature representation capabilities. Consequently, when faced with the aforementioned challenges of small objects and complex backgrounds, their performance bottleneck becomes particularly prominent, making it difficult to balance detection efficiency and accuracy.

1.4. Our Main Contributions

To address the aforementioned challenges, this paper takes YOLOv11 as a baseline to conduct an in-depth application and performance evaluation of an improved lightweight detection algorithm that integrates advanced network

structures for the task of traffic sign detection. The main contributions of this paper are as follows:

(1) This paper systematically investigates the impact of introducing a higher-resolution P2 feature layer into the detection neck of YOLOv11 on the performance of small object detection. Experiments demonstrate that this modification can structurally and significantly enhance the model's perceptual capabilities for tiny objects.

(2) This paper conducts detailed experimental comparisons of different internal configurations of an advanced integrated fusion detection module (termed FPN-P2 in this paper for ease of discussion). This module applies a "shallow feature direct-path, deep feature fusion" strategy, aiming to efficiently integrate multi-scale features.

(3) Through extensive comparative and ablation studies on the TT100K dataset, this paper, for the first time, comprehensively evaluates the performance of different module combinations under this architecture. Ultimately, it identifies the optimal model configurations in two distinct dimensions—performance and efficiency—providing valuable empirical evidence for model selection in specific scenarios.

2. Related Work

2.1. Traditional Object Detection Algorithms

Prior to the widespread adoption of deep learning, traditional object detection algorithms typically followed a multi-stage pipeline: first, selecting candidate regions using strategies like sliding windows; then, extracting handcrafted features; and finally, using a classifier for discrimination.

The research focus of this era was centered on the design of feature extractors. In 2001, the Viola-Jones detector proposed by Viola and Jones became a milestone in this field[2]. By leveraging Haar-like integral features combined with an Adaboost cascaded classifier, it achieved real-time face detection for the first time, and its concepts have had a lasting impact. Subsequently, the Histogram of Oriented Gradients (HOG) features proposed by Dalal and Triggs described object shapes by statistically analyzing gradient information in local image regions[3]. When combined with Support Vector Machines (SVMs), HOG achieved excellent results in pedestrian detection tasks. Building upon this foundation, Felzenszwalb et al. further introduced the Deformable Part Model (DPM), which treats an object as a collection of parts, allowing it to better handle non-rigid deformations and representing the pinnacle of traditional detection algorithms[13].

Despite these achievements, traditional algorithms heavily relied on handcrafted features. Such features were often designed for specific tasks, exhibiting insufficient generalization ability and robustness. Consequently, their performance would degrade sharply when faced with complex and varied real-world scenarios.

2.2. Deep Learning-based Object Detection Algorithms

Following AlexNet's tremendous success in the 2012 ImageNet competition, the powerful automatic feature learning capabilities of Convolutional Neural Networks (CNNs) brought about a revolutionary breakthrough in object detection. Deep learning-based algorithms eliminate the need for manual feature design and can learn end-to-end from data, achieving performance that far surpasses traditional methods.

Currently, the field is primarily divided into two major technical routes: two-stage and one-stage.

2.2.1. Two-Stage Detection Algorithms

Two-stage algorithms are characterized by their high accuracy, with the R-CNN series being the representative work. The R-CNN proposed by Girshick et al. was the first to introduce CNNs to object detection, but its pipeline was cumbersome and slow. Subsequent works like Fast R-CNN and Faster R-CNN made improvements, with the latter's introduction of the Region Proposal Network (RPN) being particularly significant[4]. The RPN integrated candidate box generation into the neural network, enabling end-to-end training. Despite their leading accuracy, the sequential "propose-then-detect" workflow of two-stage methods limits their detection speed, making them difficult to meet real-time requirements.

2.2.2. One-Stage Detection Algorithms

One-stage algorithms, known for their high efficiency, perform dense sampling directly on feature maps to predict object classes and locations. The YOLO (You Only Look Once) series, proposed by Redmon et al., is a paradigm of this approach[7]. The evolution began with YOLOv1, which unified the detection problem into a single regression task, and advanced to YOLOv3, which significantly improved small object detection performance by introducing multi-scale feature pyramids for prediction[6]. Subsequently, YOLOv4 and YOLOv5 built upon previous works by integrating a series of state-of-the-art optimization strategies (known as "Bag of Freebies" and "Bag of Specials"), such as the CSPNet backbone and the PANet neck structure, becoming one of the most widely adopted models in industrial applications[7,14,15].

In recent years, the development of the YOLO series has shown an accelerating trend. YOLOv8 further optimized the network architecture (e.g., the C2f module) and loss functions, adopting an Anchor-Free design to establish a new high-performance baseline[8]. Following this, YOLOv9, proposed by Wang et al. in 2024, introduced Programmable Gradient Information (PGI) to alleviate information loss in deep networks and designed a novel Generalized Efficient Layer Aggregation Network (GELAN) architecture, achieving a significant breakthrough in parameter utilization efficiency[9]. Shortly after, YOLOv10, released by a team from Tsinghua University, marked a milestone innovation by enabling end-to-end real-time detection through an NMS-Free design[10]. The YOLO ecosystem continues to evolve; for instance, GOLD-YOLO, proposed by Li et al., reconstructs the traditional feature pyramid neck by designing a novel "Gather-and-Distribute" mechanism, further enhancing the fusion efficiency of multi-scale features[11]. These latest advancements represent the cutting-edge explorations of the general-purpose object detection framework in various directions.

2.3. Research on Small Object Detection

Small object detection is a widely recognized challenge in the field of object detection due to the low pixel proportion and scarce feature information of such objects. To tackle this challenge, researchers have proposed various optimization strategies, which can be broadly categorized as follows:

(1) Multi-Scale Feature Learning: This category of methods aims to enhance the network's representation capabilities for objects of different scales. The Feature Pyramid Network (FPN), proposed by Lin et al., is a classic

work in this area[16]. It propagates high-level semantic information to low-level, high-resolution feature maps through a top-down pathway with lateral connections. Building on this, the Path Aggregation Network (PANet) added an additional bottom-up information path, further shortening the information propagation distance[15]. Subsequent works like BiFPN and NAS-FPN have explored more efficient cross-scale feature fusion topologies[17]. The introduction of the P2 high-resolution feature layer for prediction in this paper precisely follows the core idea of this technical route, which is to leverage finer-grained, low-level features to improve the perception of small objects.

(2) Introduction of Contextual Information: Since small objects have limited intrinsic features, utilizing their surrounding contextual information is an effective strategy for auxiliary judgment. In recent years, Attention Mechanisms have been widely used to weight important regions in feature maps and suppress background noise.

(3) Data Augmentation: Augmenting small object samples at the data level is another direct optimization method. For example, over-sampling images containing small objects or using methods like Copy-Paste to randomly paste small object instances onto different training images can increase their frequency and diversity during training.

It is worth noting that the idea of incorporating shallower feature layers (such as the P2 layer) into the detection head has been preliminarily explored and validated in advanced general-purpose detection frameworks like YOLOX and other small object detection domains[18]. However, the problem of how to match the newly added P2 layer with a highly collaborative and lightweight feature fusion network, and to successfully apply and systematically evaluate it in the specific context of traffic sign detection, remains an issue worthy of in-depth research.

2.4. YOLO-based Traffic Sign Detection Research

Given the outstanding performance of the YOLO series in real-time detection, numerous scholars have applied these models to the task of traffic sign detection, implementing various adaptive modifications tailored to the characteristics of this scenario. These improvement efforts can be primarily categorized into the following directions:

In terms of feature fusion and enhancement, which is the most mainstream research direction, Wang et al., for example, introduced the Convolutional Block Attention Module (CBAM) into the neck structure of YOLOv5 to address the insufficient feature discrimination capability in complex backgrounds[19]. By serially applying spatial and channel attention, they enhanced key features and suppressed noise, achieving improved performance on the CCTSDB dataset. Similarly, to enhance model robustness in adverse weather conditions, Du et al. combined the Coordinate Attention mechanism with an improved feature fusion network, effectively boosting the model's detection accuracy[20]. These studies fully validate the effectiveness of attention mechanisms in the task of traffic sign detection.

In terms of model lightweighting, to meet the deployment requirements of resource-constrained platforms such as in-vehicle systems, some research has focused on reducing the model's computational complexity while maintaining accuracy. For instance, to enable mobile deployment, Yan et al. replaced the backbone of YOLOv5 with the more lightweight MobileNetV3 and incorporated Coordinate

Attention for optimization, achieving decent detection results at a lower computational cost[21].

In terms of small object detection optimization, some researchers have recognized the issue of insufficient perception of tiny objects in traditional YOLO models. For example, Zhang et al. specifically aimed to improve the detection capability for tiny traffic signs by adding an extra prediction branch to the feature pyramid structure of YOLOv5 and refining its path aggregation method[22].

In summary, although previous works have made effective optimizations to YOLO models in various aspects, there is still room for further exploration. Most existing research has concentrated on introducing attention modules or improving fusion structures at the established P3-P5 scales, while fewer studies have addressed the problem from the more fundamental level of detection scales to solve the issue of feature loss in shallow network layers caused by overly small objects in high-resolution images. Therefore, conducting a systematic performance evaluation of an integrated, lightweight fusion detection framework that starts from the P2 layer and is specifically optimized for tiny objects represents a pressing gap to be filled in this field, and it is precisely the core entry point of this paper's work.

3. The Proposed Algorithm Model

3.1. Overall Network Architecture

This study is based on YOLOv11 and conducts an in-depth application and evaluation of an advanced detection architecture, focusing on two core technical routes. The first route involves the structural optimization of the model's feature fusion neck to assess its efficiency and performance. The second route explores the effect of introducing a higher-resolution feature scale in the detection head to enhance its perceptual capability for tiny objects. To clearly articulate this evaluation process, the subsequent sections will respectively introduce the three-scale configuration of this architecture, which serves as our efficiency-optimal solution, and the four-scale configuration, which represents our performance-optimal solution.

3.2. Analysis of the Baseline Model YOLOv11n

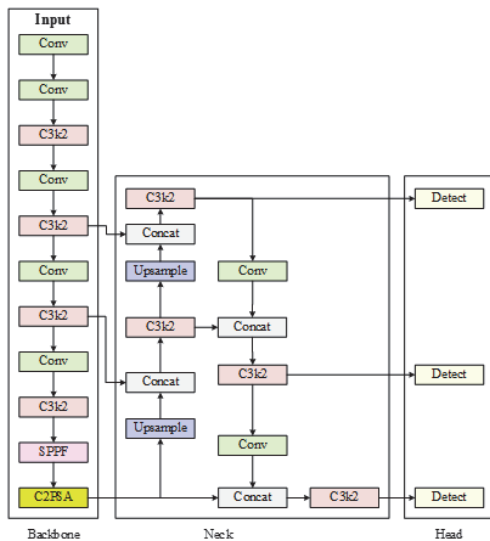


Figure 1. Baseline Model (yolo11) Network Architecture

The YOLOv11n, which this study is based on, is a lightweight detection model with excellent performance. Its

backbone employs a series of convolutional and C3k2 modules to efficiently extract image features. Its neck adopts the classic Path Aggregation Network (PANet) structure, which performs both top-down and bottom-up bidirectional fusion on the P3 (8× downsampling), P4 (16× downsampling), and P5 (32× downsampling) feature levels from the backbone output to enhance feature representation[15]. Finally, its Detect Head concurrently performs object classification and bounding box regression predictions on the three fused feature maps. The original architecture of yolo11 is shown in Figure 1:

Although YOLOv11n performs exceptionally well in general-purpose object detection tasks, its architecture exhibits inherent limitations when applied to specific, small-object-dense scenarios like traffic sign detection. Firstly, its minimum detection scale is constrained by the P3 layer. For traffic signs with extremely small pixel dimensions in the original image (e.g., smaller than 16×16 pixels), their effective features may become difficult to distinguish after 8× downsampling, leading to missed detections. Secondly, the standard PANet structure treats all scales equally during the feature fusion process and does not give sufficient attention to the shallow features that small objects rely on, which limits further accuracy improvements.

3.3. Efficiency-Optimal Model: The AFPN(C3k2) Three-Scale Detector

To evaluate the performance of advanced fusion networks on the traffic sign detection task, we first investigated a model configuration that structurally optimizes the neck and head while maintaining the P3-P5 three-scale detection paradigm. As shown in Figure 2, this model replaces the structurally separate PANet and Detect Head of the baseline with a unified, integrated module, which we term the AFPN(C3k2) module.

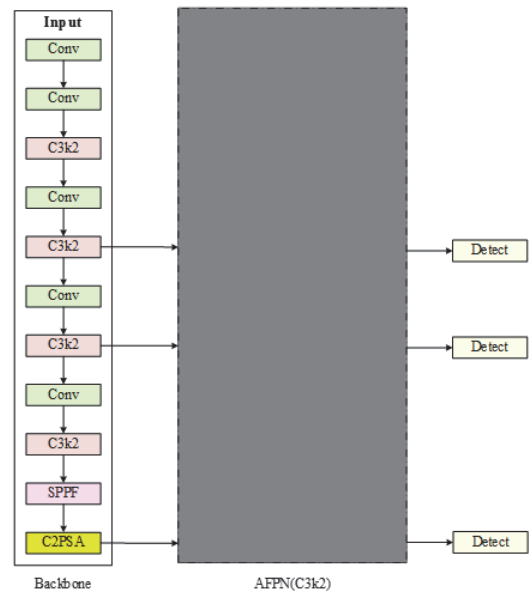


Figure 2. The yolo11-AFPN-P345-Custom Model Network Architecture

The ability of this module to achieve "cost reduction and efficiency enhancement" lies in two key improvements to the traditional PANet structure. Its internal design philosophy can be summarized as follows:

(1) Adoption of More Efficient Core Building Blocks: The neck of a traditional PANet is typically stacked with multiple

standard convolutional blocks and bottleneck structures. In contrast, the AFPN(C3k2) module exclusively utilizes the more advanced C3k2 module as its core building block. C3k2 is a module that draws inspiration from CSPNet, splitting the input feature map into two parts along the channel dimension: one part is passed directly through a shortcut connection, while the other undergoes deep processing. This "stem-residual" style architecture can significantly reduce parameter count and computational load while maintaining or even enhancing the network's feature learning capability, as it avoids redundant computations on superfluous gradient information.

(2) Optimized Cross-Scale Connections and Information Flow: In addition to upgrading the core units, the AFPN(C3k2) module also optimizes the cross-scale connection paths of the feature pyramid. Unlike the fixed upsampling and downsampling paths in a standard PANet, this module may employ fewer convolutional layers for transitions and exercises finer control over the channel numbers after fusion. This results in less loss of high-level semantic and low-level localization information during transmission, thereby achieving higher fusion efficiency.

Through the combination of these two aspects, the AFPN(C3k2) module ultimately constructs a neck network with stronger feature representation and more efficient information fusion, all while not increasing, and in fact even reducing, computational costs. As demonstrated in the subsequent ablation study (Section 4.4.2), this architectural modification is the fundamental reason why the model can still achieve a significant performance leap over the baseline without introducing the P2 layer, representing the efficiency-optimal configuration discovered in this targeted comparative study.

3.4. Performance-Optimal Model: The FPN-P2 Four-Scale Detector

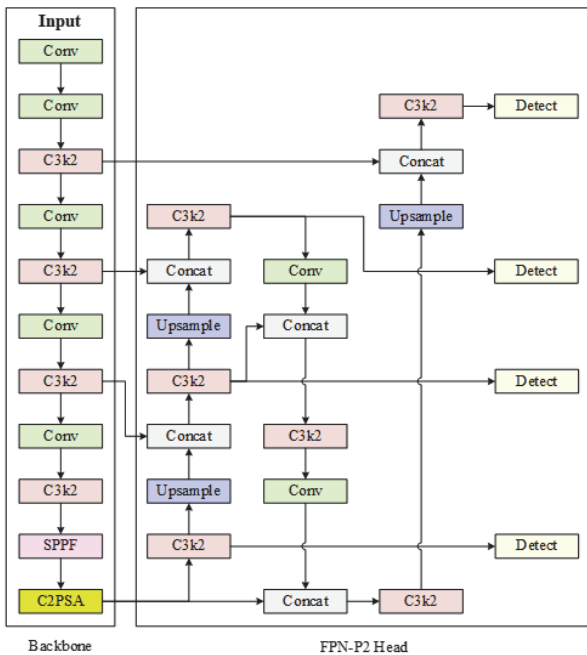


Figure 3. The yolo11-AFPN-P2345 Model Network Architecture

Building on the validated effectiveness of the advanced fusion network, we further conducted an in-depth evaluation of a model configuration that introduces the P2 detection scale to fundamentally address the issue of small object

feature loss. The detailed network architecture of this configuration, which was also identified as the performance-optimal model in this study, is shown in Figure 3.

The core of this architecture lies in its "shallow detail direct-path" designed for the P2 layer, and we collectively refer to its integrated neck and head module as FPN-P2 for ease of discussion. Its detailed working principles can be summarized as follows:

(1) Four-Way Feature Input: The FPN-P2 module directly receives inputs from four key nodes of the backbone, which are the feature map outputs of P2 (4× downsampling), P3 (8× downsampling), P4 (16× downsampling), and P5 (32× downsampling).

(2) Deep Semantic Fusion Backbone: The module first constructs a fusion backbone targeting the three mid-to-high-level features: P3, P4, and P5. This backbone follows a PANet-like structure, progressively passing and fusing high-level strong semantic information (e.g., "this is a traffic sign") down to lower-level feature maps via a top-down pathway. Subsequently, it enhances localization information through a bottom-up pathway. This step aims to ensure the model has a comprehensive understanding of medium-to-large objects and the contextual environment of the scene.

(3) Privileged Fusion Path for Shallow Details: The most significant difference between this FPN-P2 structure and the traditional PANet design lies in its handling of the P2 layer. We posit that the fine-grained spatial details carried by the P2 layer (such as the edges and textures of small objects) are extremely valuable and easily corrupted. Therefore, this structure does not involve the P2 layer in the complex deep fusion loop described above. Instead, it designs a "privileged" fusion path for it: as shown in the top part of Figure 3, the feature map from P3, after one top-down fusion and upsampling, is simply concatenated in a lightweight manner with the original P2 feature map from the backbone. This is then immediately fed into its dedicated C3k2 module for final feature integration.

(4) Four-Scale Parallel Prediction: Ultimately, the model performs parallel predictions on four scales: a P2 detection head composed of "original P2 + lightly-fused P3," which maximally preserves detail information, and three other detection heads for P3, P4, and P5 that have undergone full semantic fusion.

Through this asymmetric design, the FPN-P2 module not only ensures the model's strong semantic understanding of medium-to-large objects but also maximally preserves the original details of the P2 layer via the "privileged direct-path," achieving precise localization of tiny objects. As shown in subsequent experiments, this architecture achieved the best detection performance on the TT100K dataset.

4. Experiments and Analysis

4.1. Dataset Introduction

All experiments in this paper were conducted on the Tsinghua-Tencent 100K (TT100K), a large-scale, public benchmark dataset for traffic sign detection [12]. As one of the most comprehensive and challenging traffic sign datasets captured under naturalistic driving conditions in China to date, TT100K was jointly collected and constructed by Tsinghua University and Tencent. It comprises 100,000 high-resolution (2048×2048 pixels) real-world street view images that cover a wide variety of complex traffic scenes in Chinese cities. Following the official data partitioning standard, we

utilized 6,598 images for the training set, 1,889 images for the validation set, and 970 images for the final test set to evaluate the models' performance.

The TT100K dataset is particularly well-suited for evaluating the robustness and practical applicability of small object detection algorithms due to its inherent challenges. The main challenges are detailed as follows:

(1) Extreme Class Imbalance: The dataset is annotated with over 200 categories of traffic signs, but the sample distribution is highly uneven. As shown in Figure 4, a few common categories (e.g., speed limit signs) have thousands of instances, while a large number of categories have very few. This presents a severe "long-tail effect," which rigorously tests the generalization ability of the models. A selection of traffic sign categories is exemplified in Figure 5.

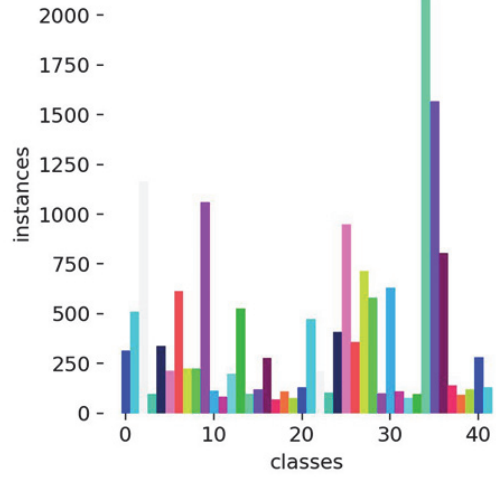


Figure 4. Distribution of instances per category in the TT100K dataset

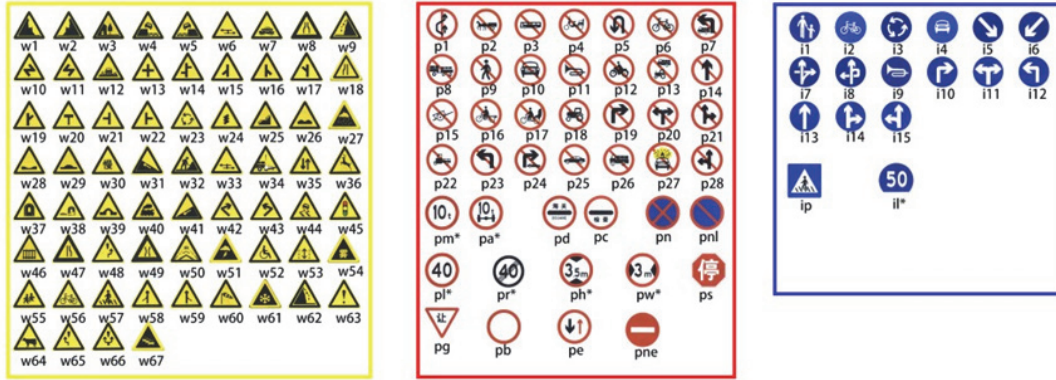


Figure 5. Examples of annotated traffic sign categories in the TT100K dataset

(2) Drastic Variation in Object Scales and Complex Scenes: In addition to class imbalance, another major challenge of TT100K is that the object size range is extremely large, with the vast majority being small objects. Furthermore, as illustrated in Figure 6, the images were captured in real-world road environments that contain various complex visual

interference factors. These include diverse lighting conditions at different times of the day (e.g., strong daylight in Fig. 6(d) and side backlighting in Fig. 6(b)), and random occlusions caused by other vehicles or roadside objects (e.g., trees in Fig. 6(b)). All these factors demand a high level of robustness from the algorithm.



Figure 6. Examples of challenging scenarios in the TT100K dataset

To enhance the model's generalization ability and effectively suppress overfitting, we employed a series of online data augmentation strategies during the training process.

These mainly included: Mosaic image stitching, MixUp augmentation, random affine transformations (such as random rotation, scaling, translation, and shearing), and color space adjustments (such as random perturbations to hue, saturation, and value). All input images were uniformly resized to 640×640 pixels before being fed into the network.

4.2. Experimental Setup and Parameters

All experiments in this paper were conducted under a unified hardware and software environment to ensure a fair comparison. The detailed configurations are as follows:

Hardware Environment: GPU: NVIDIA GeForce RTX 3090; CPU: Intel(R) Xeon(R) CPU E5-2682 v4 @ 2.50GHz; RAM: 50GB.

Software Environment: OS: Windows; Deep Learning Framework: PyTorch 1.12.0; CUDA: 11.3; Python: 3.8.20.

Training Hyperparameters: To guarantee the fairness of all comparative experiments, we applied an identical set of

training hyperparameters to all models, as detailed in Table 1.

Table 1. Training Hyperparameter Settings

Hyperparameter	Value
Input Image Size	640 x 640
Epochs	300
Batch Size	16
Optimizer	SGD
Initial Learning Rate (lr0)	0.01
Learning Rate Scheduler	Cosine Annealing
Weight Decay	0.0005

4.3. Evaluation Metrics

To comprehensively and objectively evaluate the performance of the algorithms studied in this paper, we adopted a series of widely recognized evaluation metrics from the field of object detection [23]. The primary metrics include Precision, Recall, mean Average Precision (mAP), model size (Params), and computational complexity (GFLOPs).

Precision refers to the proportion of true positives among the instances predicted as positive by the model. Its formula is shown in Equation 1:

$$P = \frac{TP}{TP+FP} \quad (1)$$

where TP (True Positive) is the number of correctly predicted positive samples, and FP (False Positive) is the number of incorrectly predicted positive samples.

Recall refers to the proportion of all actual positive samples that are successfully predicted by the model. Its formula is shown in Equation 2:

$$R = \frac{TP}{TP+FN} \quad (2)$$

where FN (False Negative) is the number of positive samples that were not predicted (i.e., missed detections).

Mean Average Precision (mAP) is the core metric for measuring the overall performance of a model. It is often calculated at an IoU threshold of 0.5 to assess a model's detection accuracy. This metric, denoted as mAP@0.5, allows for a standardized comparison of performance across different models. Its formula is shown in Equation 3. Furthermore, mAP@0.5:0.95 provides richer performance information by considering the model's average precision across a range of IoU thresholds. Its formula is shown in Equation 4:

$$mAP_{50} = \frac{1}{N} \sum_{i=1}^N AP_i(IoU=0.5) \quad (3)$$

$$mAP_{50:95} = \frac{1}{K} \sum_{k=1}^K \frac{1}{N} \sum_{i=1}^N AP_i(IoU=\tau_k) \quad (4)$$

where N is the number of classes, and τ_k represents different IoU thresholds.

GFLOPs (Giga Floating-point Operations) is a metric for the computational complexity of a model, representing the number of floating-point operations performed during a single forward pass. It quantifies the computational demand

of the model. For a standard convolutional layer, it is calculated as shown in Equation 5:

$$GFLOPs = 2HW(C_{in}k^2 + 1)C_{out} \quad (5)$$

where H, W are the height and width of the output feature map, C_{in} is the number of input channels, k is the kernel size, and C_{out} is the number of output channels.

Parameters (Params) is a metric for the size of a model, representing the total number of its weight parameters. It is closely related to the model's memory footprint and file size. For a standard convolutional layer, it is calculated as shown in Equation 6:

$$Params = (C_{in}k^2 + 1)C_{out} \quad (6)$$

4.4. Experimental Results and Analysis

To systematically validate the effectiveness of the improved model architectures studied in this paper, we conducted a series of exhaustive comparative and ablation experiments. This section provides an in-depth analysis from three perspectives: overall performance, module effectiveness, and visualization of results.

4.4.1. Main Experiment: Performance Comparison with Mainstream Lightweight Models

To objectively evaluate the performance level of the architectures studied in this paper, we selected four mainstream lightweight object detection models from both classic and cutting-edge series as baselines: YOLOv5n, YOLOv8n, YOLOv11n, and the larger-sized YOLOv11s [7,8]. All models were trained and tested under identical experimental configurations. We compared the performance-optimal configuration identified in our research against these baselines. The comprehensive performance comparison on the TT100K validation set is presented in Table 2.

Based on the experimental results in Table 2, we can draw the following conclusions:

(1) The optimal model configuration studied in this paper (yolo11-AFPN-P2345) achieves state-of-the-art (SOTA) performance across all key metrics. Its mAP@0.5 reached 76.4%, demonstrating a commanding lead of 13-14 percentage points over the three baseline models, YOLOv5n, YOLOv8n, and YOLOv11n. Furthermore, its precision (75.9%) and recall (68.7%) are also superior, indicating that the model's predictions are not only accurate but also

comprehensive.

Table 2. Comprehensive Performance Comparison of the Optimal Model and Mainstream Lightweight Models

Module	Epoch	P(%)	R(%)	mAP50(%)	mAP50:95(%)	Params(10^6)	GFLOPs
YOLOv5n	300	67.6	55.7	61.8	46.9	2.51	7.1
YOLOv8n	300	67.2	59.2	63.3	47.9	3.01	8.1
YOLOv11s	300	74.3	67.6	74.9	47.4	9.43	6.4
YOLOv11n	300	65.6	62.1	62.1	58.0	2.59	21.4
yolo11-AFPN-P345-Custom	300	70.3	62.2	68.8	52.8	2.12	6.4
yolo11-AFPN-P2345	300	75.9	68.7	76.4	58.8	2.91	19.7

(2) The studied architecture successfully breaks the performance bottleneck of existing lightweight designs. As seen in the table, the three n-series models from different technical routes (YOLOv5n, v8n, v11n) exhibit strikingly similar performance on the TT100K dataset, suggesting a common "ceiling" for general-purpose lightweight designs when tackling such small-object tasks. Our architecture, by introducing the P2 layer and an optimized fusion network, successfully elevates the performance to a new level.

(3) The optimal model achieves an excellent balance between efficiency and accuracy. It is particularly noteworthy that compared to the larger and more computationally intensive YOLOv11s, our optimal model (yolo11-AFPN-P2345), with only about one-third of its parameters and lower computational complexity, not only surpassed its mAP@0.5

but also matched its performance on the more stringent mAP@0.5:0.95 metric. This fully demonstrates the advancement and high efficiency of the architectural design investigated.

4.4.2. Ablation Study

To systematically dissect and validate the actual contributions of the core design elements within the studied architecture—namely, the AFPN fusion network and the introduction of the P2 detection scale—we designed a series of exhaustive ablation experiments. Starting from the yolo11 baseline, we conducted a comprehensive evaluation by progressively evolving the model configurations. The results are shown in Table 3.

Table 3. Ablation Study Results of Core Improvement Modules

Baseline	AFPN	P2345	P345	Custom	Epoch	P (%)	R (%)	mAP50 (%)	mAP50:95 (%)	Params (10^6)	GFLOPs
√					300	65.6	56.8	62.1	47.7	2.59	6.4
√	√		√		300	70.5	63.2	69.7	53.8	2.67	8.9
√			√	√	300	70.3	62.2	68.8	52.8	2.12	6.4
√	√	√			300	75.9	68.7	76.4	58.8	2.91	19.7
√	√	√		√	300	64.7	65.6	69.4	53.1	2.24	13.8

(Note: √ indicates that the module or setting was adopted in the experimental configuration. P2345/P345 denotes the detection scales. Custom indicates the use of C3k2 as the core internal building block.)

From the ablation study data in Table 3, we can derive an exceptionally clear and logically rigorous optimization path and a final conclusion:

(1) Validation of the AFPN Fusion Network's Effectiveness:

First, we evaluated the performance of AFPN relative to the baseline's standard PANet without changing the detection scale (P345). Comparing row 2 (Baseline+AFPN) with row 1 (Baseline), the introduction of the "default AFPN" resulted in a significant 7.6 percentage point increase in mAP@0.5 (from 62.1% to 69.7%), which demonstrates the superiority of the adopted fusion network's design philosophy. Comparing row 3 (Baseline+AFPN+Custom) with row 2, after optimizing the AFPN internally with the C3k2 module (Custom), although the mAP@0.5 slightly decreased (from 69.7% to 68.8%), both its parameter count and GFLOPs were substantially compressed, falling even below the baseline levels. This shows that the "Custom" configuration is a successful efficiency optimization strategy.

(2) The P2 Detection Scale as the Decisive Factor for Performance Breakthrough:

Next, we investigated the impact of introducing the P2 detection scale. The most critical comparison lies between row 4 (Baseline+AFPN+P2345) and row 2 (Baseline+AFPN): on the same "default AFPN" foundation, merely switching the detection scale from P345 to P2345 caused the mAP@0.5 to leap again from 69.7% to 76.4%, bringing another massive

gain of 6.7 percentage points. This irrefutably proves that for small-object-dense datasets like TT100K, directly leveraging high-resolution shallow features for prediction is the core driver for achieving a performance breakthrough.

(3) Determination of the Optimal Combination:

The entire ablation process clearly points to a single conclusion: combining the superior-performing fusion network (the default AFPN) with the most critical improvement concept (the introduction of the P2 scale) ultimately constitutes our optimal model configuration (row 4). Intriguingly, the result of row 5 (Baseline+AFPN+P2345+Custom) also corroborates our inference: the complex C3k2 module exhibits a degree of "negative interference" with the direct features of the P2 layer, and their combined effect is far inferior to the synergy between the simpler default AFPN and the P2 layer.

In summary, the ablation study not only validates the effectiveness of each of our improvement directions but also, through rigorous modular comparison, leads us to a final determination of a model architecture that is logically optimal and demonstrates the strongest performance.

4.4.3. Visualization of Results

In addition to quantitative numerical analysis, we also visualized the detection results to more intuitively demonstrate the advantages of the algorithms studied in this

paper. We selected several representative challenging scenarios from the TT100K validation set to compare the detection results of our models with those of the baseline model yolov11n and others.

(1) Performance Comparison on Extremely Small Object Detection

To validate the small object recognition capability of our performance-optimal model (yolo11-AFPN-P2345), we

selected three sets of scenarios containing extremely small objects for comparison. These objects have minimal pixel dimensions in the original images, posing a severe test to the detection capabilities of all models. The yolo11-AFPN-P2345 model was compared against YOLOv11s, YOLOv11n, YOLOv8n, and YOLOv5n, with the results shown in Figure 7.

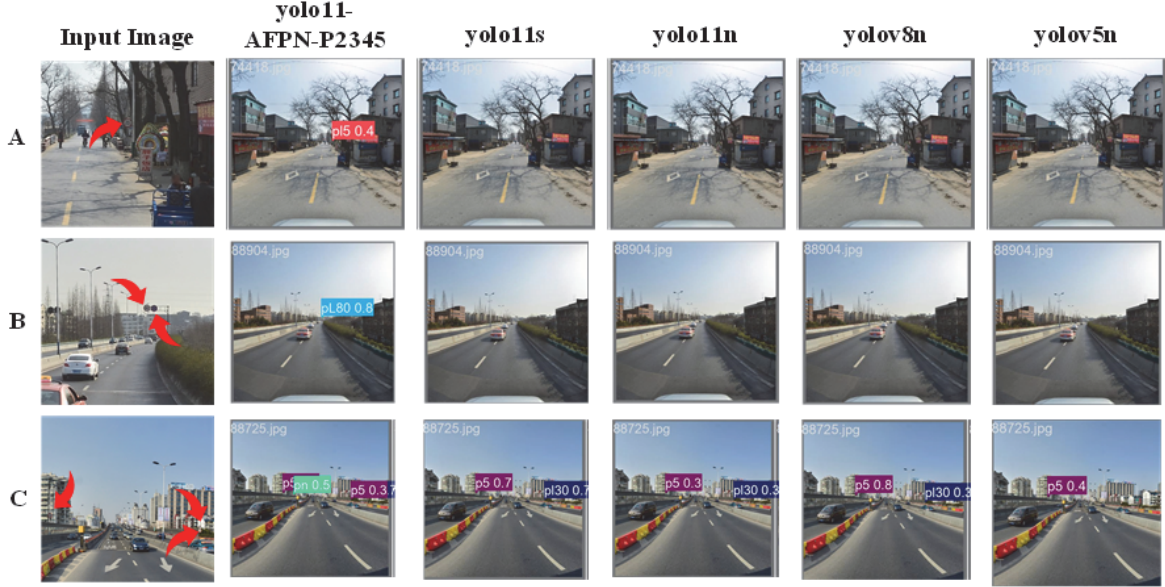


Figure 7. Comparison of the yolo11-AFPN-P2345 model and mainstream models in extremely small object detection scenarios

Group A in the figure shows a dim sign, group B shows a combination of two different signs, and group C shows three signs scattered on both sides of the photo. Red arrows point to the key targets.

A consistent phenomenon can be observed from the comparisons in groups A and B of Figure 7: for traffic signs located far in the distance with extremely sparse pixel information, both the larger YOLOv11s and other classic models suffered from complete missed detections. This is because their minimum detection scale is limited to the P3 layer, causing key information to be lost after multiple downsampling stages. The issue of missed detections is even more pronounced in group C: except for YOLOv5n which detected only one sample, the other mainstream models detected two, whereas our yolo11-AFPN-P2345 model

detected four, albeit with one false positive.

In contrast, our model, leveraging its P2 detection head, successfully captured and identified these extremely small objects in all three experimental groups. This series of visual evidence decisively proves the critical role of introducing the P2 detection scale in fundamentally solving the small object detection problem, and it is also the core reason for our model's superior performance on the mAP metric.

(2) Robustness Comparison in Complex Backgrounds

To verify the feature discrimination capability of the yolo11-AFPN-P345-Custom model, we selected a complex scenario where the traffic sign itself is small and surrounded by cluttered colors and numerous distractors. This model was compared with YOLOv11n, YOLOv8n, and YOLOv5n, with the results shown in Figure 8.



Figure 8. Comparison of the yolo11-AFPN-P345-Custom model and baseline models in complex background detection results

In the scenario shown in Figure 8, the mainstream models only detected two signs, and YOLOv5n was affected by background interference, resulting in a false positive.

In contrast, the yolo11-AFPN-P345-Custom model, leveraging the powerful feature discrimination capability of its AFPN(C3k2) fusion network, accurately locked onto the true targets while ignoring background distractors,

demonstrating optimal localization precision and anti-interference ability. This visually proves that even without changing the detection scale, the advanced fusion network we adopted possesses stronger intrinsic robustness, which is key to this model's ability to achieve high performance at a low computational cost.

In conclusion, the qualitative visualization results and the

quantitative experimental data corroborate each other, fully demonstrating the robustness and superiority of the algorithms studied in this paper when handling small, occluded, and dense object scenarios.

5. Discussion

5.1. Summary of Work

To address the bottleneck of insufficient detection accuracy faced by current mainstream lightweight object detection models in real-world traffic scenarios, which arises from small object scales and complex backgrounds, this paper has conducted an in-depth application study and a comprehensive performance evaluation of an improved YOLOv11-based lightweight detection algorithm.

The core of this research was a systematic evaluation of two levels of design: first, at the detection scale level, we validated the decisive role of introducing a higher-resolution P2 feature layer in enhancing the perception of tiny objects; second, at the feature fusion level, we compared the performance of different internal configurations of the integrated fusion module, termed FPN-P2.

Through exhaustive comparative and ablation studies on the TT100K dataset, we not only successfully validated the effectiveness of this series of improvement strategies but also revealed the multi-dimensional nature of the "optimal solution" in model design for small object detection tasks. This research ultimately led to the realization of two model paradigms, each valuable in a different application dimension:

(1) The Performance-Optimal Model (yolo11-AFPN-P2345): By combining a concise and efficient feature fusion network with the P2 detection scale, this model achieved a mAP@0.5 of 76.4% on the TT100K dataset, reaching the performance peak of this study. This result not only substantially surpasses all peer lightweight models, including YOLOv8n, but also demonstrates superior detection performance with significantly lower model complexity than YOLOv11s, setting a new benchmark for scenarios pursuing maximum accuracy.

(2) The Efficiency-Optimal Model (yolo11-AFPN-P345-Custom): This model represents the pursuit of a balance between "efficiency and accuracy." With a smaller parameter count than the baseline YOLOv11n (2.12M) and an identical computational load (6.4 GFLOPs), it achieved a high mAP@0.5 of 68.8%. Its "performance-to-cost ratio" is unmatched among all compared models. This result proves that through sophisticated network design, it is entirely possible to achieve a tremendous leap in detection accuracy without sacrificing—or even while improving—efficiency, which holds great practical significance for deployment on resource-constrained edge devices.

5.2. Limitations and Future Work

Rigorous scientific exploration is an endless endeavor. Although the work in this paper has achieved phased results, we are still aware of its existing limitations and have given deep thought to future research directions.

We believe the most significant finding of this research is the revelation of the high sensitivity of the "optimal model architecture" to the "characteristics of the specific dataset." We observed on TT100K that a simpler fusion network performed better when combined with the P2 layer, which may stem from the relatively uniform nature of the

small object features in this dataset. However, whether this conclusion is universally applicable is an open question that warrants further investigation.

Based on this, we propose the following directions for future research:

(1) Cross-Dataset Generalization Validation: Migrating the "performance-optimal" and "efficiency-optimal" models proposed in this paper to other datasets with different characteristics (such as the general-purpose objects in PASCAL VOC or the aerial small objects in VisDrone) for cross-validation. This will more profoundly reveal the true application boundaries of different network structures (e.g., simple fusion vs. complex fusion), thereby providing more valuable guidance for model selection in diverse scenarios.

(2) Exploring the Potential of Model Compression: Addressing the fact that the models studied herein still have room for optimization in terms of computational load, future work could explore the use of advanced model compression techniques such as knowledge distillation and network pruning. The aim would be to further reduce the model's inference latency while maintaining its current high accuracy, pushing it towards more extreme real-time performance.

6. Conclusion

In conclusion, this paper not only contributes two effective solutions to the field of high-precision lightweight traffic sign detection but also, through in-depth experimental exploration, provides valuable practical insights and inspiring future research paths for the broader question of "how to design an optimal network architecture based on the specific characteristics of a task."

This paper conducted an in-depth application study and comprehensive performance evaluation of an advanced YOLOv11-based architecture for traffic sign detection, focusing on solving the critical bottleneck of small object detection in lightweight models. Through a series of exhaustive comparative and ablation studies on the TT100K dataset, our research contributes two highly effective and valuable model paradigms. The performance-optimal model (yolo11-AFPN-P2345), by integrating a concise fusion network with a P2 detection scale, achieved a state-of-the-art mAP@0.5 of 76.4%, setting a new performance benchmark. Concurrently, the efficiency-optimal model (yolo11-AFPN-P345-Custom) demonstrated exceptional efficiency, achieving a high mAP of 68.8% with fewer parameters and identical computational cost as the baseline.

More significantly, our findings reveal the nuanced trade-offs between network complexity and performance in specific tasks, highlighting that the optimal architecture is context-dependent. Ultimately, this work not only provides robust solutions for high-precision lightweight traffic sign detection but also offers valuable empirical insights and inspiring research paths for the broader challenge of designing optimal network architectures tailored to specific task characteristics.

References

- [1] World Health Organization. (2018). Global status report on road safety 2018. <https://www.who.int/publications/i/item/9789241565684>.
- [2] Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Vol. 1, pp. I-511-I-518). IEEE.

- [3] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Vol. 1, pp. 886–893). IEEE. <https://doi.org/10.1109/CVPR.2005.177>.
- [4] Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>.
- [5] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 779–788). IEEE. <https://doi.org/10.1109/CVPR.2016.91>.
- [6] Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767*. <https://doi.org/10.48550/arXiv.1804.02767>.
- [7] Jocher, G. (2020). YOLOv5. *GitHub*. <https://github.com/ultralytics/yolov5>.
- [8] Jocher, G., Chilian, A., Rai, P., & Solawetz, J. (2023). Ultralytics YOLOv8. *GitHub*. <https://github.com/ultralytics/yolov8>.
- [9] Wang, C.-Y., Yeh, I.-H., & Liao, H.-Y. M. (2024). YOLOv9: Learning what you want to learn using programmable gradient information. *arXiv preprint arXiv:2402.13616*. <https://doi.org/10.48550/arXiv.2402.13616>.
- [10] Wang, A., Chen, H., Liu, L., Chen, K., Wang, Z., & He, J. (2024). YOLOv10: Real-time end-to-end object detection. *arXiv preprint arXiv:2405.14458*. <https://doi.org/10.48550/arXiv.2405.14458>.
- [11] Li, H., Li, J., Wei, H., Liu, Z., Wang, Z., & Xu, Z. (2024). GOLD-YOLO: A novel real-time object detector. *arXiv preprint arXiv:2403.11331*. <https://doi.org/10.48550/arXiv.2403.11331>.
- [12] Zhu, Z., Liang, D., Zhang, S., Huang, X., Li, B., & Hu, S. (2016). Traffic-sign detection and classification in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2110–2118). IEEE. <https://doi.org/10.1109/CVPR.2016.232>.
- [13] Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9), 1627–1645. <https://doi.org/10.1109/TPAMI.2009.167>.
- [14] Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*. <https://doi.org/10.48550/arXiv.2004.10934>.
- [15] Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 8759–8768). IEEE. <https://doi.org/10.1109/CVPR.2018.00913>.
- [16] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2117–2125). IEEE. <https://doi.org/10.1109/CVPR.2017.228>.
- [17] Tan, M., Pang, R., & Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 10781–10790). IEEE. <https://doi.org/10.1109/CVPR42600.2020.01079>.
- [18] Ge, Z., Liu, S., Wang, F., Li, Z., & Sun, J. (2021). YOLOX: Exceeding YOLO series in 2021. *arXiv preprint arXiv: 2107.08430*. <https://doi.org/10.48550/arXiv.2107.08430>.
- [19] Wang, J., Bai, Y., Zhang, Y., & Wang, H. (2022). Research on traffic sign detection algorithm based on improved YOLOv5. In 2022 IEEE 5th International Conference on Information Systems and Computer Aided Education (ICISCAE) (pp. 432–436). IEEE. <https://doi.org/10.1109/ICISCAE.55895.2022.9936214>.
- [20] Du, J., Wang, Z., Zhang, T., Ma, Y., & Li, B. (2023). A real-time traffic sign detection method based on improved YOLOv5 in bad weather. *Electronics*, 12(13), 2788. <https://doi.org/10.3390/electronics12132788>.
- [21] Yan, B., Zhang, J., Wang, S., Chen, S., & Li, P. (2023). A lightweight traffic sign detection method based on improved YOLOv5. In 2023 7th International Conference on Virtual and Augmented Reality Simulations (ICVARS) (pp. 130–134). IEEE. <https://doi.org/10.1109/ICVARS57878.2023.10134443>.
- [22] Zhang, X., Liu, C., & Zhao, Z. (2022). An improved YOLOv5-based traffic sign detection algorithm. In 2022 International Conference on Computer Technology and Information Science (CTIS) (pp. 247–251). IEEE. <https://doi.org/10.1109/CTIS.55621.2022.00057>.
- [23] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The PASCAL Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2), 303–338. <https://doi.org/10.1007/s11263-009-0275-4>.