mathematics

MDPI

Article

Enhanced YOLOX with United Attention Head for Road Detetion When Driving

Yuhuan Wu and Yonghong Wu *

School of Science, Wuhan University of Technology, Wuhan 430074, China; lovexxxand24@whut.edu.cn
* Correspondence: whyflying2008@163.com

Abstract: Object detection plays a crucial role in autonomous driving assistance systems. It requires high accuracy for prediction, a small size for deployment on mobile devices, and real-time inference speed to ensure safety. In this paper, we present a compact and efficient algorithm called YOLOX with United Attention Head (UAH-YOLOX) for detection in autonomous driving scenarios. By replacing the backbone network with GhostNet for feature extraction, the model reduces the number of parameters and computational complexity. By adding a united attention head before the YOLO head, the model effectively detects the scale, position, and contour features of targets. In particular, an attention module called Spatial Self-Attention is designed to extract spatial location information, demonstrating great potential in detection. In our network, the IOU Loss (Intersection of Union) has been replaced with CIOU Loss (Complete Intersection of Union). Further experiments demonstrate the effectiveness of our proposed methods on the BDD100k dataset and the Caltech Pedestrian dataset. UAH-YOLOX achieves state-of-the-art results by improving the detection accuracy of the BDD100k dataset by 1.70% and increasing processing speed by 3.37 frames per second (FPS). Visualization provides specific examples in various scenarios.

Keywords: object detection; feature extraction; spatial attention; attention head; loss function

MSC: 68T45

Citation: Wu, Y.; Wu, Y. Enhanced YOLOX with United Attention Head for Road Detetion When Driving. *Mathematics* **2024**, *12*, 1331. https://doi.org/10.3390/math12091331

Academic Editor: Ivan Lorencin

Received: 23 March 2024
Revised: 20 April 2024
Accepted: 22 April 2024
Published: 27 April 2024

1. Introduction

Machine learning simulates the capacity of learning patterns and features from data. With the advent of massive data and more complex application scenarios, conventional machine learning shows limitations when processing high volumes of data generated in medical, industrial, and engineering fields. Deep convolutional neural networks (DCNN) with more hidden layers have more complex network structures and more powerful feature learning and expression capabilities compared to traditional machine learning methods. Computer vision, as a field of artificial intelligence (AI), refers to the ability of computers and systems to extract meaningful information from images, videos, and other visual inputs and make decisions based on the information. DCNN algorithms have achieved remarkable results in many large-scale recognition tasks in the field of computer vision since they were proposed.

Restricted by self-reaction, drivers may be unable to accurately identify road conditions within a short time, leading to incorrect decisions and operational accidents. Benefiting from advancements in artificial intelligence (AI), the Internet of Things (IoT), and mobile communication, the assistant driving system, which comprises perception, decision-making, and execution, has gained more applications in engineering to improve traffic efficiency and driving safety [1,2]. The widespread availability of vision-based sensors facilitates the process of perception. As a crucial component, perception processes a vast number of visual data and performs various tasks, such as pedestrian detection [3], road detection [4], traffic lane detection [5], and drivable area segmentation [6]. In an assistant driving system

that handles images, real-time processing and high precision are two essential requirements to guarantee timely and accurate decision making for driving safety [7]. However, meeting both requirements in real-world application scenarios is a significant challenge due to limited computational resources and device memory. Currently, many models can achieve very high accuracy, but at the same time, the model parameters are extensive.

Object detection involves localizing and classifying targets in images, and it is a fundamental task for vision-based assistant driving systems. In recent years, the rapid advancement of DCNN has led to the emergence of numerous outstanding object detection models. For example, region convolutional neural networks (R-CNN) [8] is a two-stage object detection algorithm that initially generates proposals and then performs fine-grained object detection. This approach was later developed into Fast R-CNN [9], Faster R-CNN [10], Mask R-CNN [11], etc. On the other hand, YOLO [12] is a one-stage object detection algorithm that directly extracts features to classify and localize objects without generating region proposals. RetinaNet [13] introduces a novel Focal Loss to address the extreme foreground–background class imbalance problem during training. CenterNet [14] detects each object by using a triplet of keypoints to overcome the limitations of a large number of inaccurate object bounding boxes. EfficientDet [15] introduces a weighted bi-directional feature pyramid network (BiFPN) and a compound scaling method that uniformly scales resolution, depth, and width.

To process road scene information in a timely manner, the processing speed for detection tasks in autonomous driving applications should exceed 30 frames per second (FPS). Over the past few years, several one-stage algorithms have been developed for driving assistance, such as YOLOv3 [16], Centernet [14], Retinanet [13], etc. However, they are difficult to apply in real-world autonomous driving scenarios when using low-resolution images as input or in fast-changing traffic scenes. Therefore, models with higher accuracy and faster transmission speeds are more likely to be applied.

In this paper, we propose a novel object detection algorithm called UAH-YOLO to address the issue of the previous model being too large or unable to accurately identify the road scene. Based on YOLOX [17], we replaced the backbone network with GhostNet [18], an efficient and lightweight architecture. We designed a unified attention head before the YOLO head. A more efficient loss function, CIOU Loss [19], is utilized during training iterations.

The main contributions of this paper are as follows.

1. This paper proposes a united attention head to precede the YOLO head for extracting scale, location, and contour information. Especially, a spatial self-attention mechanism is designed to obtain spatial similarity through convolutions and pooling.
2. Our proposed algorithm, UAH-YOLO, outperforms the compared algorithms in object detection tasks on the BDD100k dataset, including YOLOX, YOLOv3, YOLOv5, EfficientDet, Faster R-CNN, and SSD. It achieves higher detection accuracy on average and has a faster processing speed, surpassing YOLOX [17] by 3.47 frames per second and far outpacing YOLOv3, Faster R-CNN, etc.
3. The UAH-YOLO algorithm has been demonstrated to be a superior detection algorithm on the Caltech Pedestrian dataset, accurately identifying pedestrians on both sides of the road when driving.

The rest of the paper is organized as follows. Section 2 presents the benchmark model used in this paper and related works on object detection. Section 3 introduces the proposed improvement strategies in detail. Section 4 presents the results of comparison experiments, ablation studies, and visualization of the BDD100k dataset. Section 5 presents the conclusions of the paper and suggests directions for future research.

2. Related Work

2.1. Methodolgy

YOLOX [17] represents an advancement in the YOLO series. As a high-performance detector, YOLOX employs advanced detection techniques, such as anchor-free detection, a

decoupled head, and the SimOTA label assignment strategy. The architecture of YOLOX is depicted in Figure 1. Similar to the original YOLO, YOLOX consists of three components: a backbone feature network, a feature pyramid network (FPN), and a YOLO head as a detector.

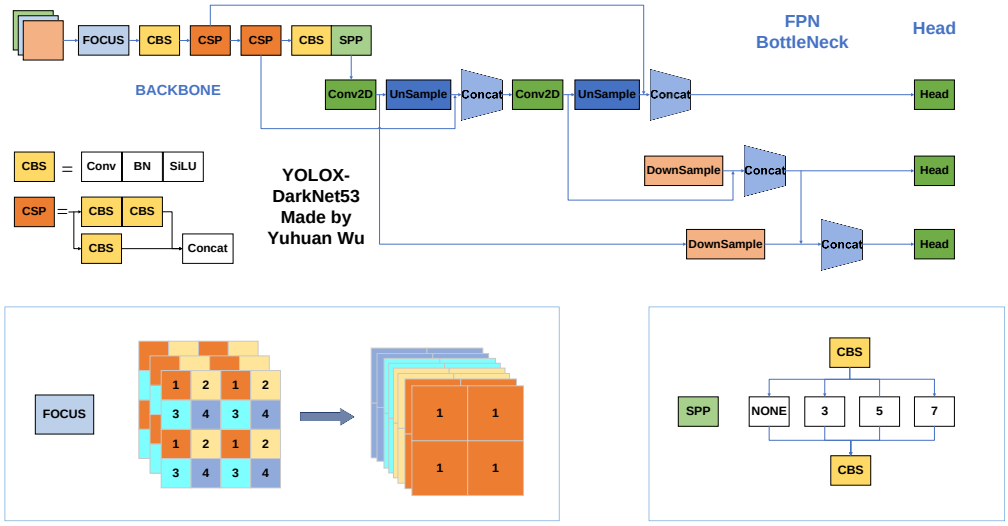


Figure 1. The architecture of YOLOX-DarkNet53.

The backbone feature extraction network utilized by YOLOX is CSPDarkNet [20], a residual network. YOLOX follows the focal network structure utilized in YOLOv5. It achieves this by obtaining four independent feature layers through the selection of every other pixel. These feature layers are then stacked to increase the input channels fourfold. The input is processed through a 1×1 convolution and a 3×3 convolution and then added to the residual component to produce the output. CSPNet [21] is employed in the backbone network of YOLOX to enhance the learning capacity of convolutional neural networks and reduce computational costs. YOLOX follows the SPP network architecture used in YOLOv5, which extracts features by pooling with kernels of different sizes to improve the network's receptive field.

The Feature Pyramid Network (FPN) is designed to generate high-level semantic feature maps at all scales. The architecture is top-down with lateral connections. A significant amount of work has been completed prior to the FPN. Featurized image pyramid, as a conventional approach, involves resizing images to various scales and computing features independently on each scale for prediction. This method was widely utilized during the era of hand-engineered features. YOLOv1 [12] and YOLOv2 [22], as well as Faster R-CNN [10], use a single feature map to enable faster detection. However, this approach leads to subpar detection performance for small targets. SSD [23] utilizes the pyramidal feature hierarchy computed by a ConvNet [24], enabling predictions at multiple scales. However, there is a lack of fusion between features. The Feature Pyramid Network (FPN), proposed by Tsung-Yi Lin et al., has demonstrated significant improvement as a versatile feature extractor in numerous applications. The FPN consists of a bottom-up pathway, a top-down pathway, and lateral connections. The bottom-up pathway computes forward through ConvNet and generates a hierarchical feature structure consisting of feature maps at multiple scales. The top-down pathway upsamples feature maps that are rich in semantic information and acquires a feature structure at the same level. Lateral connections merge feature maps of the same size from both the bottom-up and top-down pathways. The structure of FPN [25] is depicted in Figure 2a.

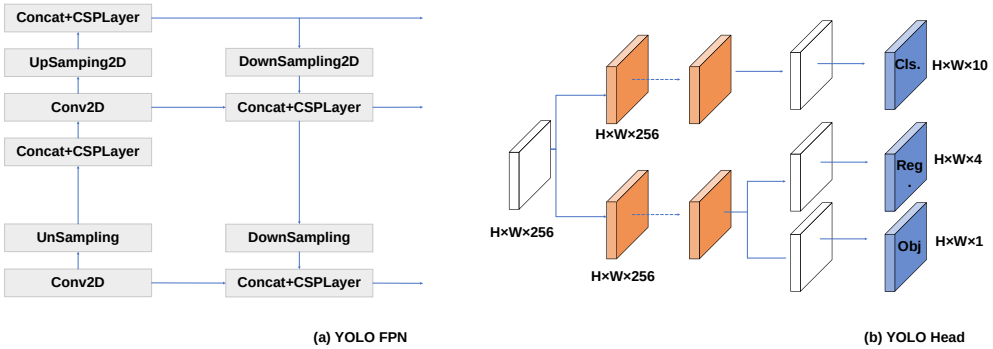


Figure 2. The architecture of YOLO FPN and YOLO Head.

Differently from the previous YOLO head, the YOLOX [17] head detector implements classification and regression in separate convolutional layers. Three features obtained by FPN are integrated into the YOLO head for prediction. The structure of the decoupled head is illustrated in Figure 2b. The "Reg" part is used to determine the regression parameters for each feature point and adjust the prediction boxes. The "Obj" part is used to determine whether each feature point corresponds to an object. The "Cls" part is used to determine the type of object.

2.2. Object Detection Research

Significant research has been conducted in recent years on object detection in assistant driving scenarios. Min Yang proposed a target recognition algorithm based on the Faster R-CNN algorithm and a Multi-Strategy Regional Proposal Network (MSPRN) [26] to optimize the anchor boxes. Yingfeng Cai et al. proposed an effective object detector, YOLOv4-5D [27], based on YOLOv4 to improve detection accuracy while enabling true real-time operation. YOLOv4-5D introduces a new feature fusion module called PAN++ and presents an optimized network pruning algorithm, resulting in a 4.23% improvement on the BDD100K dataset. This approach is well-suited for real-time object tracking. Mingyuan Sun et al. proposed YOLO-I [28] for detecting infrared targets on roads, replacing the conventional structure and improving feature extraction using the advanced EfficientNet [29]. Meanwhile, the k-means algorithm and data augmentation strategy were implemented, achieving a mean average precision of 0.89 with an average detection speed of 10.65 frames per second. Shuqi Fang et al. [30] enhanced the Mask R-CNN by replacing the ResNet backbone network with the ResNeXt network, which includes group convolution, and by incorporating an efficient channel attention module (ECA) to the backbone feature extraction network. The proposed models achieved a mean average precision of 62.62% for target detection and 57.58% for segmentation accuracy on the publicly available CityScapes autonomous driving dataset. These results were 4.73% and 3.96% better, respectively, than those achieved by Mask R-CNN.

3. Proposed Methods

3.1. Backbone

While CSPDarkNet extracts features and achieves excellent object detection, the large model parameters limit the deployment of YOLOX on mobile devices with varying resource constraints. Autonomous driving computing platforms require the simultaneous processing of multiple sensors and computing tasks, including detection, tracking, and decision making. Therefore, it is essential to investigate portable and efficient network architectures to conserve memory and computational resources.

The parameters of a convolutional neural network mainly come from the convolution kernel. Therefore, determining the number and size of the convolution kernel is crucial for controlling the size of the model. The formula for calculating parameters in a convolutional layer can be expressed as follows, where C_{Out}, C_{In}, K_w, K_h denotes the output channels, in-

put channels, width of kernels, height of kernels, respectively. Deep separable convolution reduces the parameters by decreasing the input and output channels.

$$Params = C_{Out} \times C_{In} \times K_w \times K_h + bias \tag{1}$$

The redundancy in feature maps is an important characteristic of successful CNNs. GhostNet generates multiple ghost feature maps by applying a series of inexpensive operations to a set of intrinsic feature maps. To reduce the number of parameters and calculations, this paper replaces CSPDarkNet with GhostNet as the backbone network for feature extraction.

Firstly, the Ghost module obtains the intermediate feature maps $F_1 \in \frac{C_{Out}}{2} \times H \times W$ through regular convolution, and F_1 are passed through deep separable convolutions to obtain $F_2 \in C_{Out} \times H \times W$, also known as inexpensive operations, which involve a small number of parameters. Then, the two feature maps F_1, F_2 are concatenated along the channels to obtain the outputs. The Ghost bottleneck, composed of Ghost modules, is divided into the main part and the residual part. The structures of the Ghost module and Ghost bottleneck are shown in Figure 3a,b.

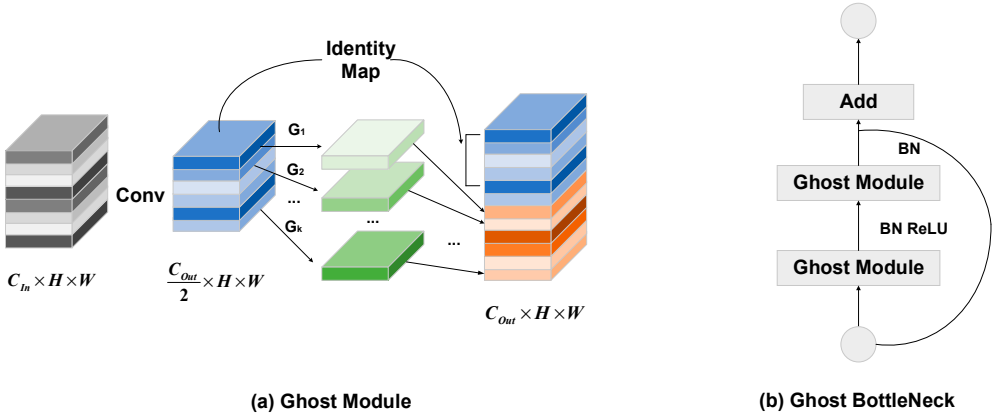


Figure 3. The architecture of Ghost Module and Ghost Bottleneck.

3.2. United Attention Head

Attention mechanisms in deep learning enable neural networks to automatically learn and selectively focus on important information when processing input data, thereby improving the performance and generalization ability of the model. However, it is challenging for a single convolution to simultaneously make the model focus on different features of the data. In this paper, we propose a united attention head which focuses on three dimensions: scale-awareness, position-awareness, and contour-awareness, denoted as ϕ_S, ϕ_P, ϕ_C, respectively.

3.2.1. Scale-Aware

Scale-aware attention is introduced to address the issue of scale invariance in object detection. Given the feature tensor $F \in R^{L \times C \times H \times W}$, where L denotes the number of layers for the input of pyramid, C represents the number of channels, H represents the number of height, and W represents the number of width, respectively. Additionally, $S = H \times W$ is defined. Thus, the input features are shaped as three-dimension tensors $F \in R^{L \times C \times S}$. The formulation of scale-aware attention is:

$$\phi_S(F) = \sigma(f(\frac{1}{SC}\sum_{S,C} F)) \cdot F \tag{2}$$

where $f(\cdot)$ is a linear function approximated by a 1×1 convolutional layer and $\sigma(\cdot)$ is a hard-sigmoid function.

3.2.2. Position-Aware

The backbone network focuses on feature extraction but often overlooks the location information of targets, leading to the neglect of occluded or edge targets in the image. In this section, we propose a novel attention mechanism called spatial self-attention, which is a position-aware attention mechanism.

Spatial self-attention obtains the query and key by convolution, which is performed in width and height, and operates average pooling, respectively. Then, it calculates the query and key by matrix multiplication and softmax activation to obtain the spatial similarity matrix, which is shaped as $C \times H \times W$. In this way, the model can identify the relationships between objects in the image. Unlike the transformer block, the input matrix is performed by convolution to obtain the value matrix and add the similarity matrix rather than dot multiplication. The attention module is depicted in Figure 4.

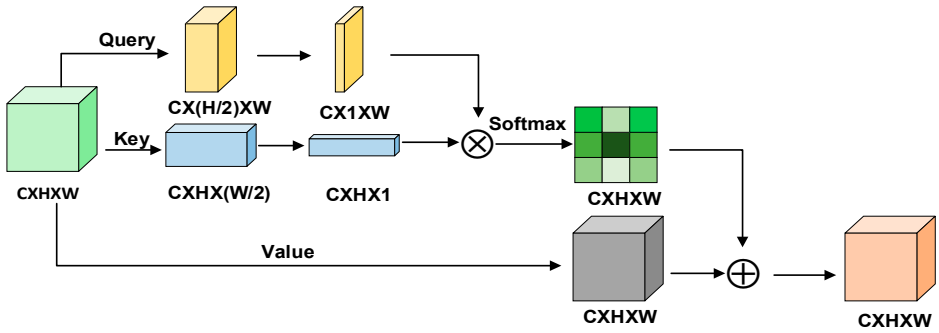


Figure 4. The architecture of spatial self-attention.

3.2.3. Contour-Aware

Traditional models like AlexNet [31], VGGNet [32], and ResNet [33] are proficient at learning features well, but they struggle to accurately distinguish between objects and backgrounds, which fails to effectively capture object contours. This section proposes a contour-aware attention mechanism to help the model focus on spatial locations and level features.

Considering a self-learned spatial offset, deformable convolution kernels shift at the sampling points of the input features, which helps the model focus on the region of interest. Contour-aware attention initially learns sparsity through deformable convolution and then aggregates features across levels at the same location:

$$\phi_C(F) = \left(\frac{1}{L} \sum_{l=1}^{L} \sum_{k=1}^{K} w_{l,k} \cdot F(l; p_k + \Delta p_k; c) \cdot \Delta m_k \right) \cdot F \tag{3}$$

where L is the number of layers, K is the square of kernel size, Δp_k is the self-learned offset to focus on the discriminative region, $p_k + \Delta p_k$ is the shift location by offset, and Δm_k is the importance scalar to be learned at the location p_k.

Not establishing an independent attention layer, we load this contour-aware attention into the spatial self-attention, replacing the traditional convolution. As a summary, the architecture of object detection with our proposed attention head is illustrated in Figure 5.

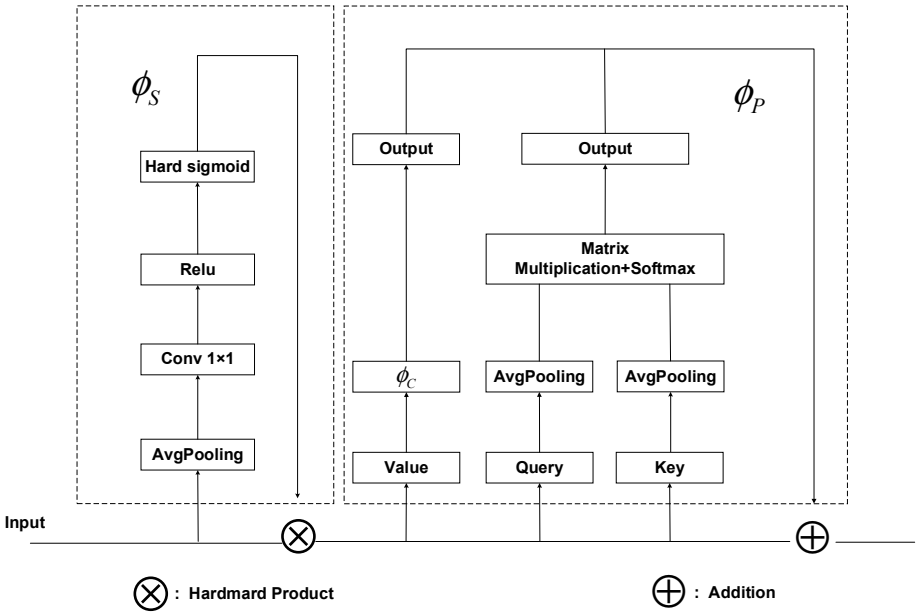


Figure 5. The architecture of United Attention Head.

3.3. CIOU Loss

The loss of YOLOX consists of three parts: $Loss_{reg}$ calculates the intersection of union between the prediction boxes and the ground truth boxes of feature points; $Loss_{obj}$ calculates the cross entropy (CE) loss between the positive and negative samples, where positive samples correspond to feature points containing objects; $Loss_{cls}$ calculates the CE loss between the prediction classes and ground truth labels of the targets. All the above definitions are calculated as follows:

$$Loss_{reg} = -\log(IOU(B_{gt}, B_{pred})) \tag{4}$$

$$Loss_{obj} = -\log(1-p) \tag{5}$$

$$Loss_{cls} = -\sum_{i=1}^{n}(t_i \log(p_i) + (1-t_i)\log(1-p_i)) \tag{6}$$

where B_{pred}, B_{gt} are the prediction boxes and ground truth boxes, p is the prediction possibility of existing objects, p_i is the prediction possibility of every class, and n is the number of all classes.

IOU Loss is designed to solve the problems of mutual independence and non-scale invariance of smooth L_1 Loss, but it is unable to reflect the magnitude of overlap. In this paper, we replaced IOU Loss with CIOU Loss to calculate $Loss_{reg}$.

$$Loss_{reg} = -\log(IOU(B_{gt}, B_{pred})) + \frac{\rho^2(b_{gt}, b_{pred})}{c^2} + \alpha\nu \tag{7}$$

$$\alpha = \frac{\nu}{(1-IOU)+\nu} \tag{8}$$

$$\nu = \frac{4}{\pi^2}(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h})^2 \tag{9}$$

where ρ denotes the distance of center points, c denotes the diagonal distance of the minimum closure region that can simultaneously contains both prediction boxes and ground truth boxes, α denotes the trade-off parameter, and ν denotes the consistency of aspect ratio.

3.4. Our Model

The architecture of our proposed model, UAH-YOLO, based on YOLOX, is illustrated in Figure 6. We replace the CSPDarkNet with GhostNet which is of lower computational complexity and add the united attention head before the YOLO head to obtain the scale, position, and contour information of the targets.

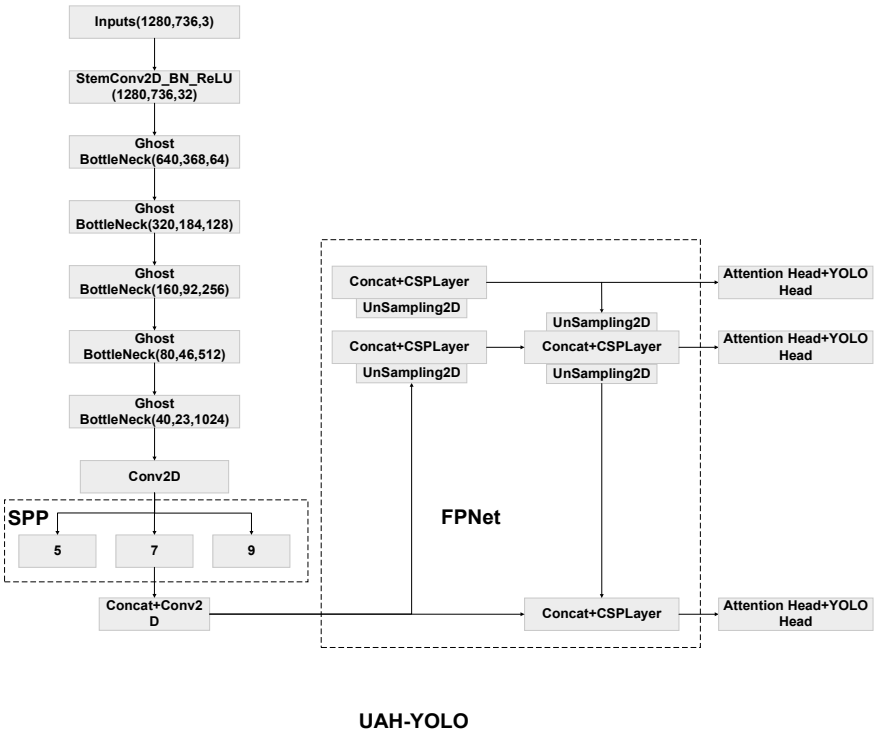


Figure 6. The architecture of UAH-YOLO.

4. Result and Discussion

4.1. Implementation Details and Evaluation Metrics

The experiments are performed under windows system, 56 GB memory, CPU: 10 vCPU Intel Xeon Processor (Skylake, IBRS), GPU: V100-32GB, PyTorch 1.9.0, and CUDA 11.1. We use stochastic gradient descent (SGD) with momentum = 0.937, wight decay = 5×10^{-4}, initial learning rate = 0.01, minimum learning rate = 1×10^{-4}, and cosine learning rate schedule. The total epochs of BDD100k and Caltech Pedestrian are 100 and 50, respectively, and the batch size is 16 by default. The training process is from scratch where some basic data augmentations are used: random crop, mosaic, mixup.

The evaluation indexes used in this paper are *Precision* (*P*), *Recall* (*R*), AP_{50}, $AP_{50–95}$. *Precision* denotes the proportion of correctly predicted true samples among predicted true samples. Recall denotes the proportion of correctly predicted true samples among positive samples. *AP* is the area under the *P*–*R* curve. *TP*, *TN*, *FP*, and *FN* are the numbers of true positive samples, true negative samples, false positive samples, and false negative samples, respectively. AP_{50} is the *AP* when the intersection of union (IOU) of the predicted box and the real box is above 50%. mAP_{50} is the average AP_{50} of all classes.

$$precision = \frac{TP}{(TP + FP)} \tag{10}$$

$$recall = \frac{TP}{(TP + FN)} \tag{11}$$

$$AP = \int_0^1 PdR \tag{12}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{13}$$

4.2. Experiment Datasets

The BDD100k and Caltech Pedestrian datasets are chosen for checking the performance of models. As a huge, complete dataset containing various weather conditions, places, and times of the day, as well as wide ranges of light conditions, occlusion, and cropping, the BDD100k dataset is normally used for the computer vision tasks in autonomous driving. The BDD100k dataset consists of 10 classes: car, truck, motor, bike, person, train, rider, bus, traffic sign, and traffic light. Considering the imbalance instances and real-world driving conditions, this paper only selects car, truck, bus, person, traffic sign, and traffic light as the investigators. Considering the fact that the dataset is too large, 10,000 images are selected for training 100 epochs. Caltech Pedestrian dataset contains approximately 10 h of 640 × 480 30 Hz video, mainly shot by small cars driving on rural streets. The video consists of about 250,000 frames, 350,000 bounding boxes, and 2300 pedestrian annotations. In this paper, we only research on the pedestrian detection tasks. Algorithm 1 shows the training process.

Algorithm 1: Without Proposal Boxes, UAH-YOLO Trains Backbone, FPNet, Attention Head for Targets Location, Detection, Classification.

Input: Target neural network Ω with parameters groups;
$\Theta = \left\{\theta_{obj}, \theta_{cls}, \theta_{reg}\right\}$;
Training set Γ;
Threshold for convergence: *thr*;
Loss function: *Loss*;
Output: Well-trained network
1: **procedure** Train (Ω, Γ)
2: **repeat**
3: sample a mini-batch (x, y) from training set Γ
4: $l \leftarrow Loss((x;\Theta), y)$
5: $\Theta \leftarrow \text{argmin}_{\Theta} l$
6: **until** $l < thr$
7: **end procedure**
8: **return** Trained network $\Omega(x;\Theta)$

4.3. BDD100k Comparison Analysis

Table 1 shows a comparison of the results on the BDD100k training sets with other state-of-the-art object detection models, including YOLOX, YOLOv3, YOLOv5, EfficientDet, Faster R-CNN, and SSD. UAH-YOLO achieved the best detection results in car, person, traffic sign, and traffic light, while YOLOX detected truck and bus better. UAH-YOLO improved the AP_{50} of car, traffic sign, traffic light, and person by 1.16%, 1.49%, 1.83%, and 0.78%, respectively. As a two-stage model, Faster R-CNN did not perform well in detecting targets and processing speed. SSD is inferior to other models like YOLOX, YOLOv3, and YOLOv5 as a one-stage model.

Table 1. Comparison of detection accuracy on BDD100k dataset—denotes the FPS is less than 10.

Method	Backbone	Car	Traffic Sign	Traffic Light	Truck	Bus	Person	mAP_{50}	FPS
UAH-YOLO	GhostNet	**0.7604**	**0.6364**	**0.5985**	0.5989	0.6048	**0.5760**	**62.90%**	**40.05**
YOLOX	CSPDarkNet53	0.7488	0.6215	0.5802	**0.6050**	**0.6072**	0.5682	61.20%	36.68
YOLOv3	DarkNet	0.6327	0.5641	0.4784	0.5421	0.5589	0.5031	54.85%	32.8
YOLOv5-s	CSPDarkNet53	0.7254	0.6032	0.5638	0.5788	0.5836	0.5612	58.68%	34.84
EfficientDet	EfficientNet	0.7158	0.6012	0.5704	0.5802	0.5794	0.5608	58.98%	30.28
Faster R-CNN	vgg-16	0.5926	0.5086	0.4462	0.495	0.5081	0.4629	51.57%	10.07
	ResNet50	0.6012	0.5011	0.4517	0.4812	0.5114	0.4598	51.92%	-
SSD	vgg-16	0.5057	0.4021	0.3766	0.4081	0.4129	0.4012	43.04%	-

Bold values indicate the best result.

Requiring a processing speed higher than 30 FPS for object detection in assistant driving applications, only UAH-YOLO, YOLOX, YOLOv3, YOLOv5, and EfficientDet can be chosen in practical application, denoting that the FPS is less than 10. Experimental results show that UAH-YOLO detected targets at the fastest speed. In conclusion, UAH-YOLO has more potential to be applied in driving assistance.

4.4. Ablation Studies

In this section, we design ablation experiments to illustrate the effectiveness of our proposed methods. We denote the backbone replacement as G, the loss function replacement as C, and the united attention head as A. The evaluation metrics mAP_{50} and parameters were chosen (Table 2).

Table 2. Ablation study on the effectiveness of each proposed method in YOLOX.

G.	A.	C.	$\mathbf{mAP_{50}}$	Parameters
×	×	×	61.20	9.92
√	×	×	61.46	**6.85**
×	√	×	61.63	10.22
×	×	√	61.87	9.97
×	√	√	61.98	10.27
√	×	√	62.68	6.90
√	√	×	62.42	7.15
√	√	√	**62.90**	7.20

Bold values indicate the best result; √ denotes the strategy is used; × denotes the strategy is not used.

The experimental results indicate that replacing the backbone network of GhostNet reduces the parameters by 3.07 MB, making it suitable and convenient for deployment on mobile devices for driving assistance. CIOU Loss function slightly increases the parameters but improves the mAP_{50} by 0.67%. The addition of attention mechanisms before the YOLO head improves the mAP_{50} by 0.43%. Overall, the detection performance of the model with three improvement methods is the most superior, with mAP_{50} increasing by 1.70% compared to the original model and reducing the parameters by 2.72 MB. Figure 7 demonstrates the prediction precision of car, person, bus, traffic light under various score threshold, respectively.

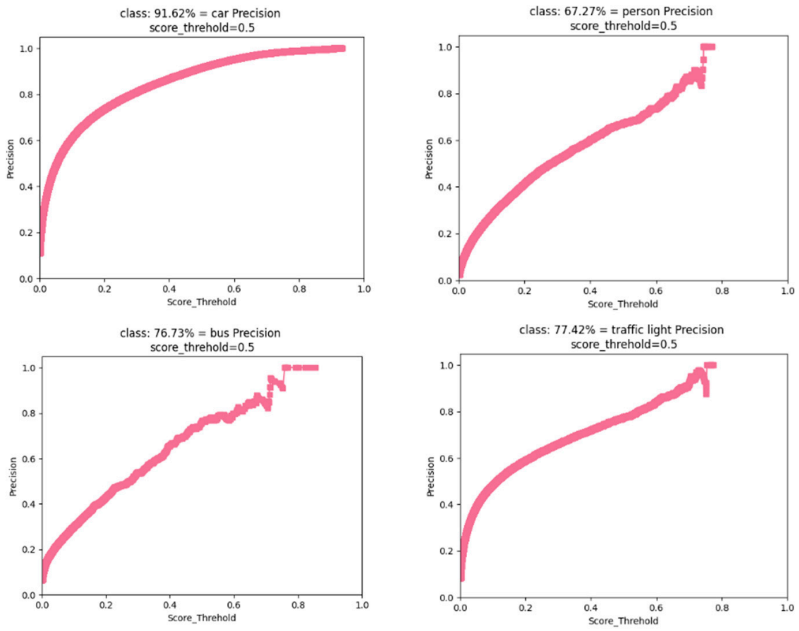
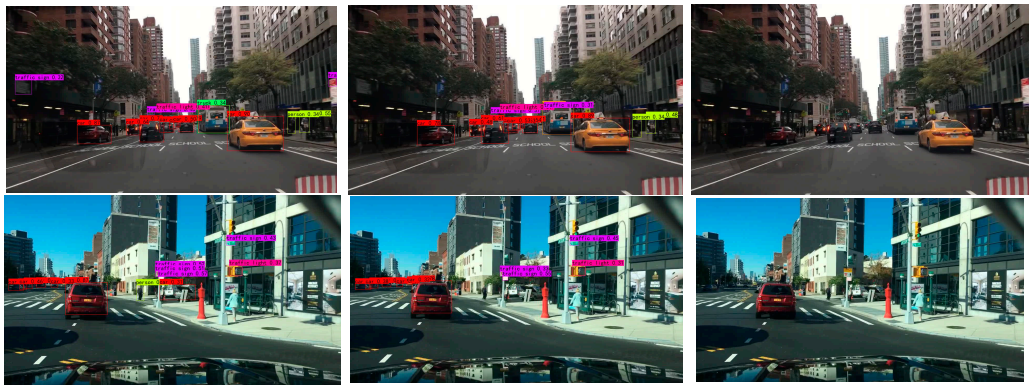


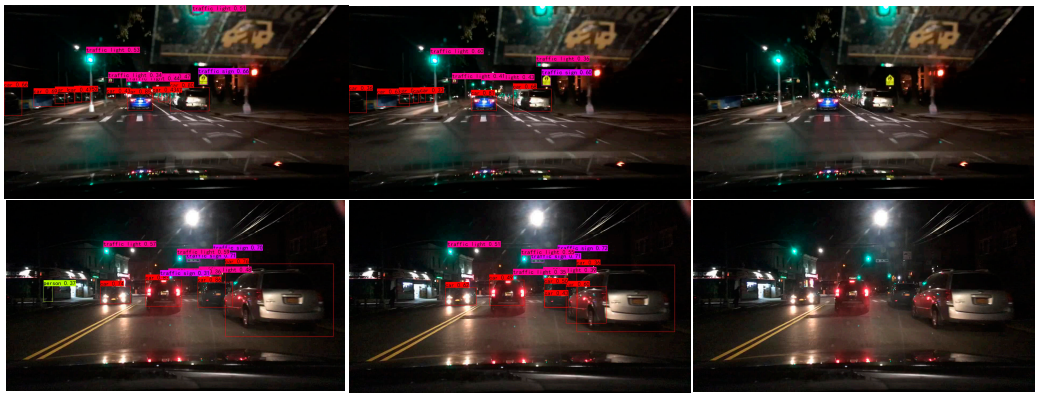
Figure 7. The precision diagram of car, person, bus, and traffic light under the score_threshold = 0.5.

4.5. Visualization of BDD100k Detection Results

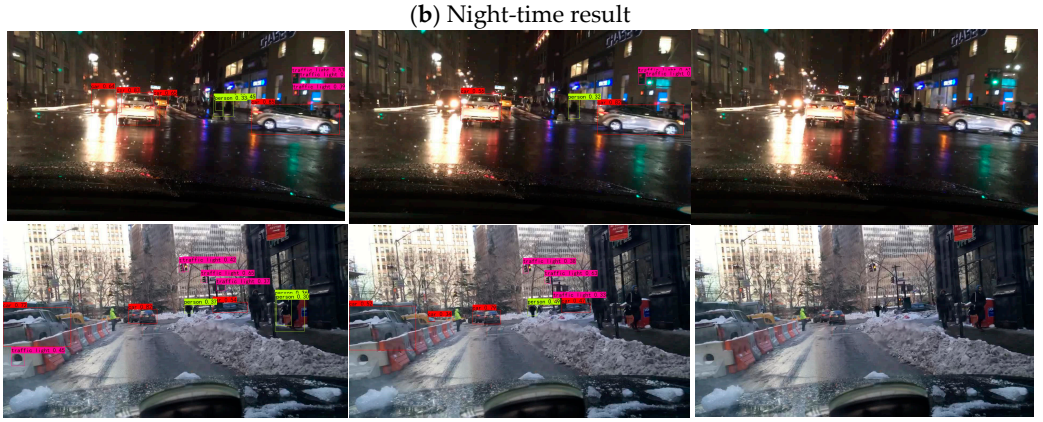
For a visual evaluation of UAH-YOLO and YOLOX, Figure 8a–c demonstrates the detection results of the UAH-YOLO and YOLOX for the BDD100k validation datasets, respectively. The detection threshold is 0.5, and the confidence level is also 0.5. We selected scenes from day, night, rainy, and snowy days for object detection. The visualization comparative analysis demonstrates that the proposed UAH-YOLO significantly outperforms YOLOX in detection tasks.



(**a**) Day-time result



(**b**) Night-time result



(**c**) Rainy and snowy result

Figure 8. Visualization of the traffic object detection results of UAH-YOLO. (**a**) Detection results in day-time scenes. (**b**) Detection results in night-time scenes. (**c**) Detection results in rainy and snowy scenes.

Figure 8a shows that during the day-time scene, UAH-YOLO demonstrates higher accuracy in detecting cars, while pedestrians on the roadside are also identified. Figure 8b demonstrates that UAH-YOLO accurately detects traffic signs, traffic lights, and cars even in dark scenes with low visibility. Figure 8c demonstrates the excellent detection ability of

UAH-YOLO when rain covers the car windows. Figure 8c demonstrates that UAH-YOLO can be utilized in complex snowy scenarios.

4.6. Train on Caltech Pedestrian Dataset

To verify the superiority of UAH-YOLO, the Caltech Pedestrian dataset was selected for pedestrian detection tasks. Sequence images were converted into JPG format, and 50 epochs were conducted where we used traditional loss function and smooth loss function, respectively. The experimental results are shown in Figure 9. As indicated in Table 3, the detection performance of UAH-YOLO is excellent. Under the condition of IOU = 0.50, the average precision reached 90.3%; under the condition of IOU = 0.50–0.95, the average recall rate reached 56.6%. As shown in Figure 10, UAH-YOLO can accurately detect pedestrians on both sides of the road, effectively preventing traffic accidents caused by pedestrians suddenly appearing.

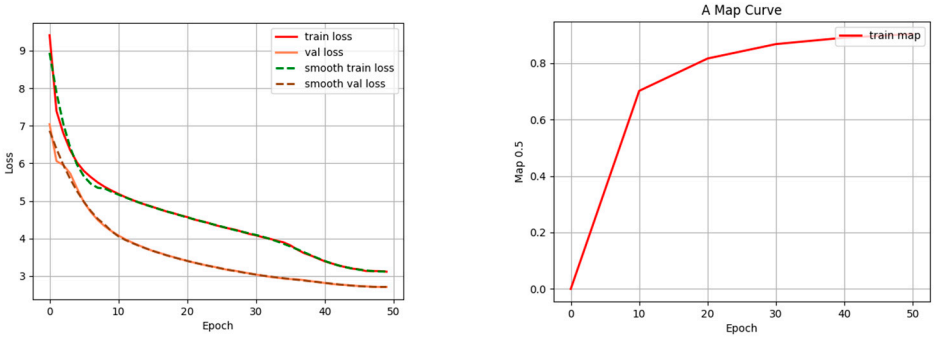


Figure 9. Training results of the Caltech Pedestrian dataset performed by UAH-YOLO.

Table 3. Pedestrian detection results of UAH-YOLO on the Caltech Pedestrian dataset.

Metrics	AP_{50}/%	$AP_{50–95}$/%	$AR_{50–95}$/%
UAH-YOLO	90.30	50.90	56.60

Figure 10. Visualization of the pedestrian on both sides of the road.

5. Conclusions and Future Scope

In this paper, we propose a novel, simple, and efficient network called UAH-YOLO, including scalable and small-sized backbone architecture, a united attention head for the scale, position, and contour information of targets, and refined objective loss function. Our model performs outstandingly on the BDD100k dataset, either approaching or surpassing the state-of-the-art results in all classifications. Meanwhile, UAH-YOLO processes images at the fastest speed. Furthermore, we have confirmed high performance in other autonomous driving detection tasks, such as pedestrian detection. In conclusion, UAH-YOLO shows great potential for deployment in the field of autonomous driving for detection tasks.

Although the attention mechanism, which is a module for feature extraction, has initially shown promising results in object detection tasks, there are still some pressing issues

that need to be addressed: the low resolution of input image pixels, failure in deep feature extraction, a lack of fusion of individual features, and the model being excessively large and lacking generalization. Object detection can be improved by optimizing the quality of input images, designing feature extraction networks, developing feature aggregation networks, and creating lightweight networks in the future.

Author Contributions: Methodology, project administration, supervision, writing—original, software, draft preparation, Y.W. (Yuhuan Wu); data curation, validation, writing—review and editing, Y.W. (Yonghong Wu). All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Natural Science Foundation of Hubei Province (No. 2020CFB546), National Natural Science Foundation of China under Grants 12001411, 12201479, and the Fundamental Research Funds for the Central Universities (WUT: 2021IVB024, 2020-IB-003).

Data Availability Statement: Publicly available datasets were analyzed in this study. The BDD100k dataset can be found at http://bdd-data.berkeley.edu (accessed on 30 May 2018). The Caltech Pedestrian dataset can be found at https://data.caltech.edu/records/f6rph-90m20 (accessed on 20 June 2009).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Yurtsever, E.; Lambert, J.; Carballo, A.; Takeda, K. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access* **2020**, *8*, 58443–58469. [CrossRef]
2. Furda, A.; Vlacic, L. Enabling safe autonomous driving in real-world city traffic using multiple criteria decision making. *IEEE Intell. Transp. Syst. Mag.* **2011**, *3*, 4–17. [CrossRef]
3. Gawande, U.; Hajari, K.; Golhar, Y. Pedestrian detection and tracking in video surveillance system: Issues, comprehensive review, and challenges. In *Recent Trends in Computational Intelligence*; Books on Demand (BoD): Norderstedt, Germany, 2020; pp. 1–24.
4. Yang, X.; Li, X.; Ye, Y.; Lau, R.Y.; Zhang, X.; Huang, X. Road detection and centerline extraction via deep recurrent convolutional neural network U-Net. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7209–7220. [CrossRef]
5. Ko, Y.; Lee, Y.; Azam, S.; Munir, F.; Jeon, M.; Pedrycz, W. Key points estimation and point instance segmentation approach for lane detection. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 8949–8958. [CrossRef]
6. Chan, Y.C.; Lin, Y.C.; Chen, P.C. Lane mark and drivable area detection using a novel instance segmentation scheme. In Proceedings of the 2019 IEEE/SICE International Symposium on System Integration (SII), Paris, France, 14–16 January 2019; pp. 502–506.
7. Teichmann, M.; Weber, M.; Zoellner, M.; Cipolla, R.; Urtasun, R. Multinet: Real-time joint semantic reasoning for autonomous driving. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1013–1020.
8. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
9. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
10. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28: Proceedings of the Annual Conference on Neural Information Processing Systems 2015, Montreal, QC, Canada, 7–12 December 2015*; MIT Press: Cambridg, MA, USA, 2015; Volume 44.
11. He, K.; Gkioxari, G.; Doll'ar, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
12. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
13. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Doll'ar, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
14. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint triplets for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6569–6578.
15. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
16. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
17. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430.

18. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1580–1589.
19. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybern.* **2021**, *52*, 8574–8586. [CrossRef] [PubMed]
20. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
21. Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 390–391.
22. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
23. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part I 14. Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
24. Tran, D.; Ray, J.; Shou, Z.; Chang, S.F.; Paluri, M. Convnet architecture search for spatiotemporal feature learning. *arXiv* **2017**, arXiv:1708.05038.
25. Lin, T.Y.; Doll'ar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
26. Yang, M. Research on vehicle automatic driving target perception technology based on improved MSRPN algorithm. *J. Comput. Cogn. Eng.* **2022**, *1*, 147–151. [CrossRef]
27. Cai, Y.; Luan, T.; Gao, H.; Wang, H.; Chen, L.; Li, Y.; Sotelo, M.A.; Li, Z. YOLOv4-5D: An effective and efficient object detector for autonomous driving. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 4503613. [CrossRef]
28. Sun, M.; Zhang, H.; Huang, Z.; Luo, Y.; Li, Y. Road infrared target detection with I-YOLO. *IET Image Process.* **2022**, *16*, 92–101. [CrossRef]
29. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning (PMLR), Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
30. Fang, S.; Zhang, B.; Hu, J. Improved mask R-CNN multi-target detection and segmentation for autonomous driving in complex scenes. *Sensors* **2023**, *23*, 3853. [CrossRef]
31. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems, Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012*; Curran Associates Inc.: Red Hook, NY, USA, 2012.
32. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
33. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.