electronics

MDPI

Article

A Vehicle Recognition Model Based on Improved YOLOv5

Lei Shao, Han Wu, Chao Li * and Ji Li

School of Electrical Engineering and Automation, Tianjin University of Technology, Tianjin 300384, China
* Correspondence: liton@email.tjut.edu.cn

Abstract: The rapid development of the automobile industry has made life easier for people, but traffic accidents have increased in frequency in recent years, making vehicle safety particularly important. This paper proposes an improved YOLOv5s algorithm for vehicle identification and detection to reduce vehicle driving safety issues based on this problem. In order to solve the problems of a disappearing model training gradient in the YOLOv5s algorithm, difficulty in recognizing small objects and poor recognition accuracy caused by the boundary frame regression function, it is necessary to implement a new function. These aspects have been enhanced in this article. On the basis of the traditional YOLOv5s algorithm, the ELU activation function is used to replace the original activation function. The attention mechanism module is then added to the YOLOv5s algorithm's backbone network to improve the feature extraction of small and medium-sized objects. The CIoU Loss function replaces the original regression function of YOLOv5s, thereby enhancing the convergence rate and measurement precision of the loss function. In this paper, the constructed dataset is utilized to conduct pertinent experiments. The experimental results demonstrate that, compared to the previous algorithm, the mAP of the enhanced YOLOv5s is 3.1% higher, the convergence rate is 0.8% higher, and the loss is 2.5% lower.

Keywords: deep learning; vehicle detection; YOLOv5; attention mechanism; artificial intelligence

Citation: Shao, L.; Wu, H.; Li, C.; Li, J. A Vehicle Recognition Model Based on Improved YOLOv5. *Electronics* **2023**, *12*, 1323. https://doi.org/10.3390/electronics12061323

Academic Editor: Eva Cernadas

Received: 8 February 2023
Revised: 6 March 2023
Accepted: 8 March 2023
Published: 10 March 2023

1. Introduction

In recent years, with the rapid development of China's industrial modernization, the number of Chinese automobiles has far surpassed the initial development of the industry. However, the frequency of traffic accidents has made the issue of safe driving one of the major research foci. Increasing attention has been paid to the development of Advanced Driver Assistance Systems (ADAS) [1] in an effort to reduce the number of accidents. ADAS systems primarily evaluate and predict the driving environment of vehicles by combining a number of sensors; in the event of a hazardous situation, the signal can be transmitted to the driver in a timely manner to ensure safe driving. Increasing numbers of people are becoming devoted to the research and development of ADAS systems as society evolves. Current ADAS systems include numerous subsystems, including Forward Collision Warning (FCW) [2]. The FCW system is an important functional component of the ADAS system, providing warning messages when a potential collision hazard is imminent, thereby preventing or reducing the severity of accident-related damage. Computer vision technology can now use advanced algorithms to detect, identify, and track objects in video [3–7] as a result of the ongoing research into computer vision by domestic and international researchers in recent years. Vehicle detection technology is a vital component of the system, and at present computer vision is primarily used to detect domestic and international targets. Using various advanced algorithms, computer vision identifies and detects objects in video [8–10].

In 2012, the proposal of the AlexNet [11,12] network sparked a new wave of deep learning algorithms, which became the predominant object detection algorithms at that time. Since then, improved object detection algorithms such as Fast RCNN [13,14], Faster RCNN [15], and R-FCN [16] have emerged. The accuracy of these proposed algorithms

has reached the optimal level, but in some instances the recognition speed falls short of the requirements. In 2016, Redmon J. proposed the YOLO [5] algorithm to improve calculation speed and ensure calculation accuracy. In the same year, the SSD [17] (Single Shot Multibox Detector) algorithm based on VGG16 (Visual Geometry Group Network) was proposed to achieve multi-scale Feature Map prediction. The algorithm employs the feature layer to detect and enhance YOLO's inadequate detection of small targets. In 2018, the Redmon J. team improved YOLOv2 [18] and obtained YOLOv3 [19] algorithm, enhanced YOLO's inadequate detection of small targets.

Zhang Fukai et al. [20] enhanced the YOLOv3 algorithm to detect vehicles. Wang Fujian et al. [21] accomplished the enhancement of the YOLO algorithm dataset's target detection. By screening VOC datasets, Ding Bing et al. [22] improved the YOLOv3 algorithm and implemented the detection of parking in highway tunnels. On the basis of the concept of transfer learning, Fu Jingchao et al. [23] enhanced the adjustment learning strategy of YOLO to improve its target detection capability. YOLOv4 [24] and YOLOv5 [25] were born in 2020. The speed and accuracy of image recognition have been significantly enhanced, and the size of the YOLOv5 model has been reduced, allowing for improved detection results in the current environment. This paper employs the YOLOv5 algorithm as its starting point for vehicle target detection.

YOLOv5's engineering practicability has improved with each iteration of the YOLO series, making it the most widely used target detection algorithm at present. According to model size, YOLOv5 is available in four variants: YOLOv5s, YOLOv5m, YOLOv5l and YOLOV5x. The only difference between the Backbone and the Neck and Prediction settings is the model's depth and width settings. More feature maps are available the deeper the backbone network, and a deeper network is more complex. In addition, the YOLOV5s network has the narrowest depth feature map width and the fastest processing speed. This paper proposes an enhanced vehicle detection algorithm based on YOLOV5s, which improves the detection accuracy of small targets and accelerates the convergence rate in response to the issues of low detection accuracy and the gradient disappearance of small targets.

2. Materials and Methods

2.1. Development of Experimental Datasets

Currently, the most popular databases for vehicle detection are the KITTI database, the general dataset VOC and COCO dataset, and the general dataset VOC. In order to improve the applicability of the model, this paper combines the KITTI open-source dataset and Internet-collected road images to create a traffic target dataset. The dataset's format is VOC, and it contains images captured from various viewing angles and orientations. Figure 1 is a schematic representation of a portion of the dataset. This article selects the three dataset categories of Car, Van, and Trunk.

Python and Qt are used to develop labeling tools. When labeling datasets, rectangular boxes are used to frame vehicles and vehicle information is noted. The precise labeling procedure is depicted in Figure 2.

Once the annotation is complete, you must use the split.py file for classification, followed by the txt2yolo_label.py file to finish the conversion from .xml to .txt. You need to use the split.py file for classification, and then use the txt2yolo_label.py file to complete the conversion from .xml to .txt. The five values represent object-class, x_center, y_center, width and height attributes. In the end, 5000 images were used for training. The experiment has a training set of 4500 and a test set of 500. The ratio of the two sets was 9:1, with approximately 12,000 vehicle targets.

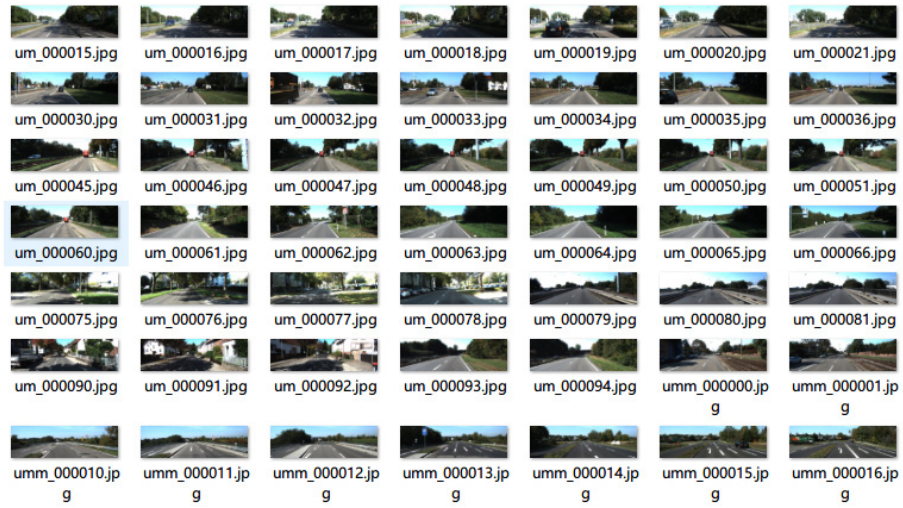


Figure 1. Portioned images extracted from the dataset.

Figure 2. Vehicle identification interface.

2.2. YOLOv5s Network Design

The YOLOV5s model is an improvement over its predecessor. The adaptive anchor frame is utilized, initially. In the training process, an expected frame is created to roughly estimate the target's position, which is then compared to the actual frame. The coordinate algorithm is used to iteratively calculate their difference. Based on this calculation, reverse update is conducted. As depicted in Figure 3, the initial predicted anchor coordinates of YOLOv5 can be obtained after multiple iterations.

```
anchors:
  - [10,13, 16,30, 33,23]  # P3/8
  - [30,61, 62,45, 59,119]  # P4/16
  - [116,90, 156,198, 373,326]  # P5/32
```

Figure 3. Initial predicted coordinates for the anchor box.

YOLOv5s will optimize the algorithm so that the network's backbone can adapt to various image inputs. Before training, the majority of algorithms will, thus, unify and standardize the input images. For instance, the image size can be scaled or expanded to the

sizes that YOLO uses most frequently, which significantly reduces the interference caused by the picture's unnecessary information to the running speed.

Additionally, YOLOv5s includes the CBL module, the Focus module, the SPP module, and the CSP module. It firstly performs convolution, batch standardization, and activation functions, which are then transferred to the Focus module for slicing processing, thereby minimizing the loss of image data. Then, it performs downsampling and the SPP module combines all parts, integrates the extracted features, and sends them to the CSP module for integration processing. The YOLOv5 network model is depicted in Figure 4.

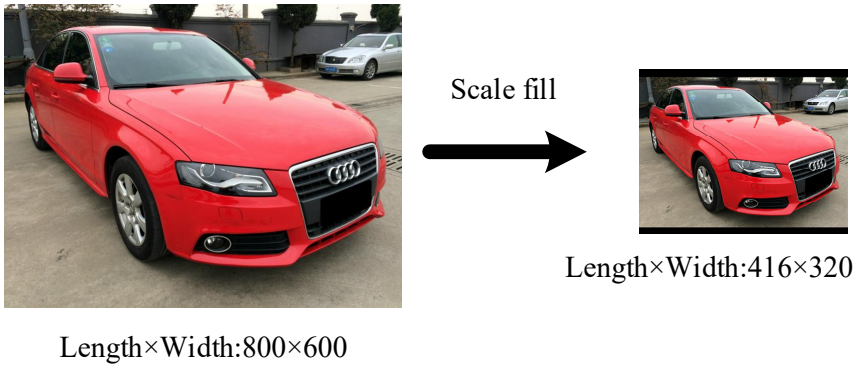


Figure 4. Size drawing after scaling.

YOLOv5s also adds CBL, Focus, SPP and CSP modules to the previous version. The CBL module mainly carries out convolution, batch standardization and function activation, and then gives it to the Focus module for slicing processing, which will greatly reduce the loss of picture information. Next, it carries out down sampling. SPP module pools all parts, fuses the extracted features, and finally sends it to CSP module for integration processing. The network model of YOLOv5 is shown in Figure 5.

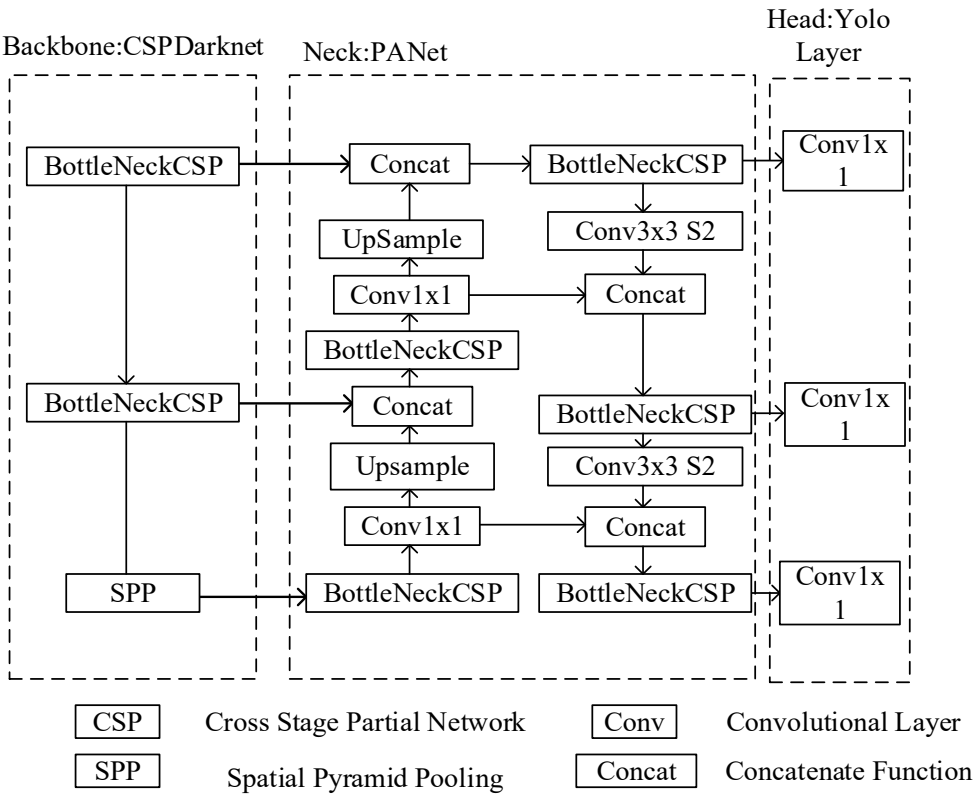


Figure 5. YOLOv5 network model diagram.

3. Methods

Due to the fact that the YOLOv5s algorithm is suitable for deployment on embedded devices with limited memory, while also meeting the accuracy requirements of the algorithm during driving and the algorithm's response speed, the YOLOv5s algorithm is currently a popular object detection algorithm. However, the YOLOv5s algorithm has numerous drawbacks: (1) YOLOv5s combines multiple activation functions in the activation function section. When multi-activation functions are combined, the training model will exhibit gradient disappearance and other issues that will further reduce its accuracy. (2) YOLOv5s has trouble identifying small objects that require identification; therefore, the algorithm's precision must be improved. (3) When a particular case exists between the detection box and the prediction box, the convergence speed of the loss function is slowed. Based on the aforementioned issues with the YOLOv5s algorithm, this chapter is based on the YOLOv5s method for vehicle detection. The activation function of YOLOv5s is first replaced. The attention mechanism module is then introduced to the backbone network in order to improve the extraction of features by YOLOv5s. The algorithm's loss function is optimized, utilizing complete intersection ratio function. Experiments were carried out to examine the algorithm's performance before and after its enhancement.

3.1. Activation Function Improvements

The CSP module of the original YOLOv5s used the Leaky ReLU function [26] and the Mish function as activation functions. When these two activation functions are utilized concurrently, the gradient will gradually diminish during back propagation and may eventually disappear. The Exponential Linear Units (ELU) activation function replaces the Leaky ReLU function and Mish function to tackle this issue. The formula for calculating the ELU activation function is depicted in the figure:

$$ELU(s) = \begin{cases} x & x > 0 \\ \alpha(e^x - 1) & x \leq 0 \end{cases} \quad (1)$$

The ELU function curve is depicted in Figure 6.

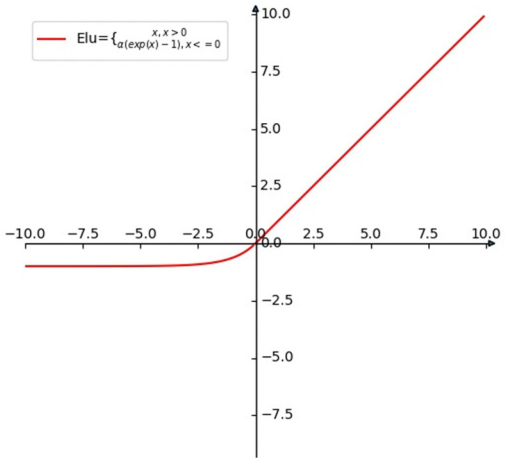


Figure 6. ELU function diagram.

The ELU function has a better linear distribution on the right side of the coordinate axis than the Leaky ReLU function, which effectively mitigates the disadvantage of gradient descent of the Leaky ReLU function. The left side of the coordinate axis is nonlinear, which may improve noise input robustness. In order to demonstrate the benefits of ELU function in a more intuitive manner, this activation function is compared to other activation functions in the COCO dataset and the results are presented in Table 1.

Table 1. Comparison table of activation functions under the COCO dataset.

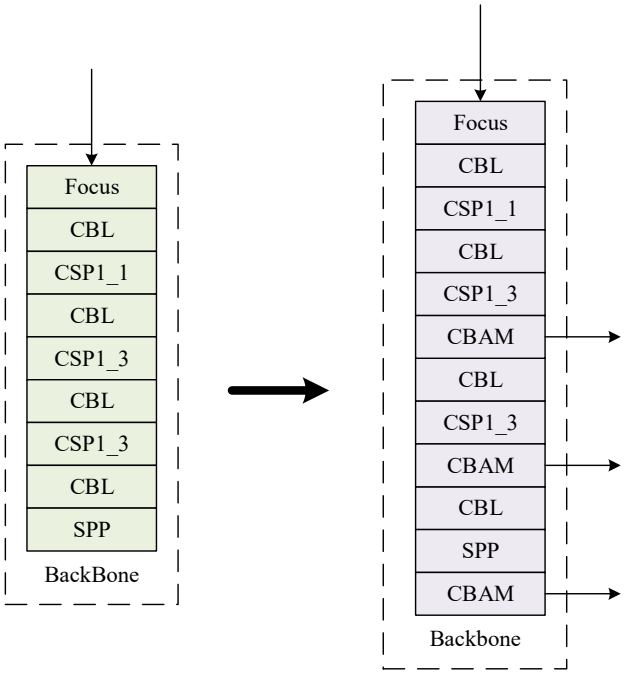
Mosaic	Label Smoothing	Leaky ReLU	Mish	ELU	Top-1 Err (%)	Top-5 Err (%)
√	√	√			22.4	5.8
√	√	√	√		21.5	5.4
√	√			√	21.0	5.1

Table 1 demonstrates that the first and fifth error rates decreased by 0.9% and 0.4% when the Leaky ReLU function and Mish function were combined as compared to the Leaky ReLU function alone, and that the first and fifth error rates decreased by 0.5% and 0.3% when the ELU function was utilized alone. According to the experimental findings, it is possible to achieve the gradient descent caused by the combination of the two activation functions. The enhanced activation function can decrease the error rate and increase the calculation's precision.

3.2. Enhanced Attention Mechanism Module

The attention process resembles the attention mechanism used by humans for object recognition. The primary information is gained by allocating sufficient resources. Important data are collected and retrieved using a convolutional neural network, which significantly enhances the precision of data collecting. The attention mechanism module may typically be added to the backbone network, and the module's parameters are simple to alter, which significantly improves the model's performance. Currently, the attention mechanism is primarily separated into two types: a channel attention mechanism represented by SE [27] (Squeeze and Excitation) and a spatial attention mechanism represented by the Convolutional Block Attention Module (CBAM [28]).

In this paper, CBAM modules were added to three main parts of YOLOv5, as shown in Figure 7. In Figure 7a, the module is added to CSP1_3(feature fusion); in Figure 7b, the CBA module is added to the Neck part of YOLOv5s after the Concat layer; in Figure 7c, the CBMA module is added before the convolution of YOLOv5's prediction module.



(**a**)

Figure 7. *Cont.*

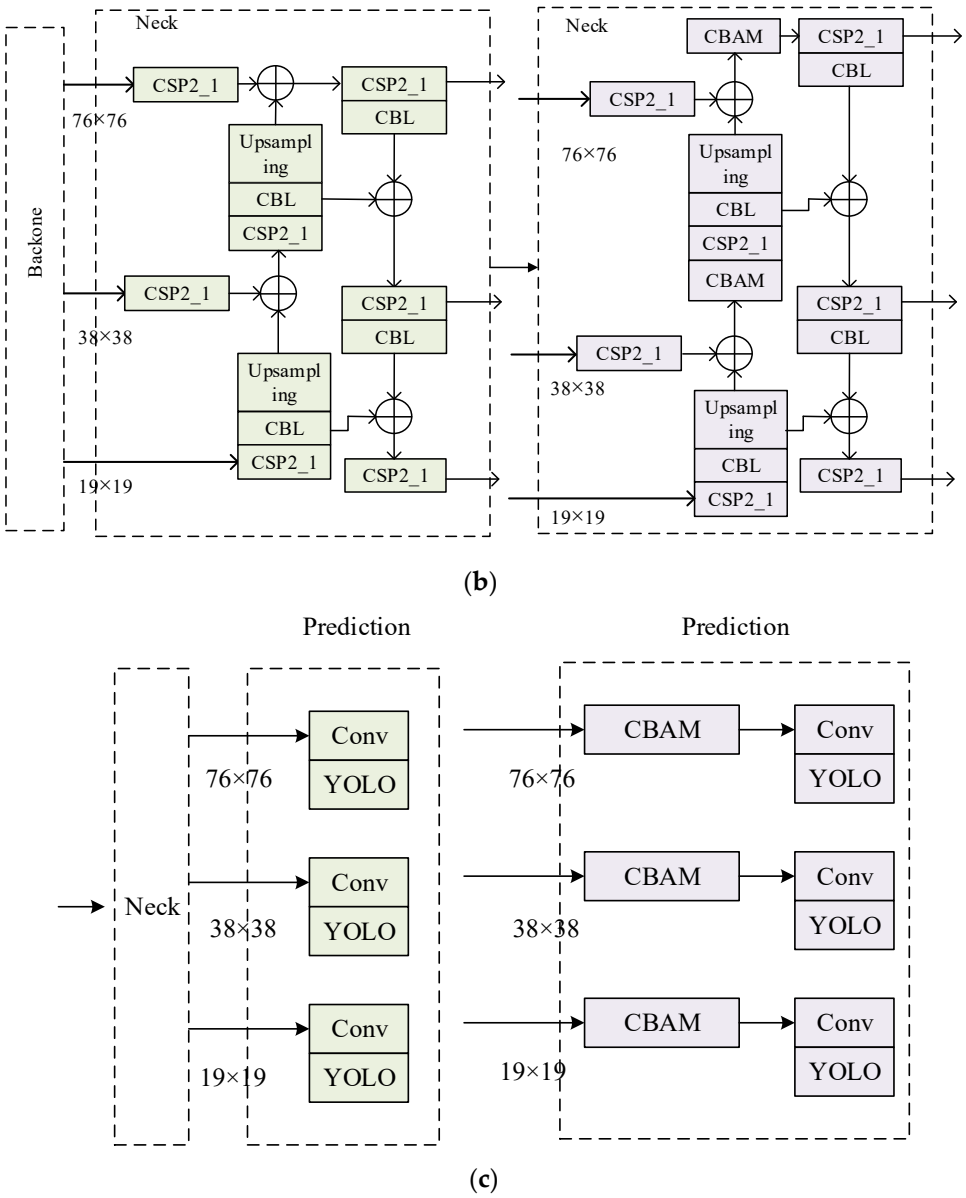


Figure 7. Network comparison before and after introducing CBMA attention mechanism. (**a**) CBAM YOLOv5s-Backbone, (**b**) CBAM YOLOv5s-Neck, (**c**) CBAM YOLOv5s-Prediction.

The comparison results of three CBAM modules in different positions and unfused YOLOv5s are shown in Table 2.

Table 2. Comparison of CBAM modules after fusion.

Network Model	AP 50%			P (%)	R (%)	mAP (%)
	Small Goal	Medium Goal	Big Goal			
YOLOv5s	83.0	97.9	99.3	76.4	92.5	92.7
CBAM_YOLOv5s-Backbone	90.4	98.2	99.4	81.2	93.8	94.1
CBAM_YOLOv5s-Neck	80.3	96.4	99.0	71.7	93.7	91.6
CBAM_YOLOv5s-Prediction	82.7	97.1	99.1	75.9	92.8	92.4

As can be seen from the table, not every fusion mode's accuracy is improved after CBAM module fusion is performed on different components of YOLOv5s. When CBAM modules are integrated into Backbone, the detection capability of small targets is greatly

improved and mAP is increased by 1.4%. Since the semantic information in Backbone networks is not rich, CBAM is added to these modules to improve the accuracy. However, for Neck and Prediction, there is no improvement in accuracy. Therefore, this document adds the CBAM module to Backbone.

3.3. Improvement of CIoU Loss Function

An Intersection over Union (IoU) [29] is typically utilized to calculate the location relationship between the predicted and actual boxes in target detection using the following formula:

$$IoU = \frac{A \cap B}{A \cup B} \tag{2}$$

As depicted in Figure 8, the original IoU formula includes several weaknesses that have been rectified. In Figure 8a, when there is no intersection between the prediction box and the real box, the result of IoU computation is 0, impeding further training and algorithm execution. In Figure 8b,c, when the prediction box is the same size as the actual box, the IoU calculation yields the same result; therefore, no judgment can be formed.

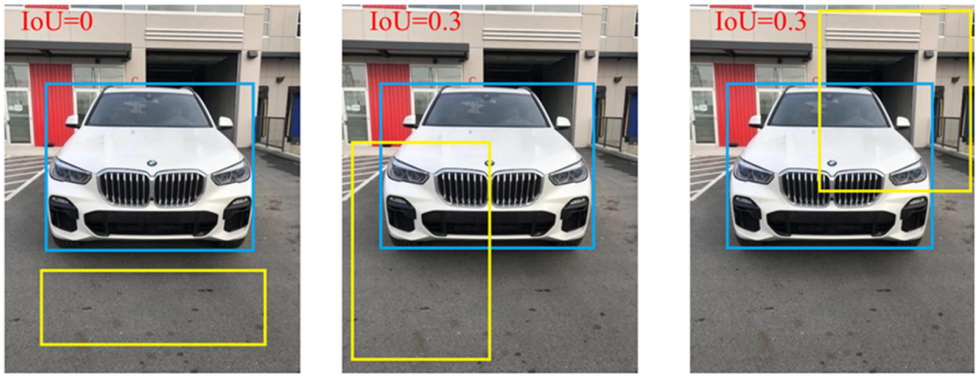


Figure 8. Network comparison before and after introducing the CBMA attention mechanism.

Therefore, in this paper, GIoU [30] (Generalized Intersection over Union) is used instead of IoU, In Figure 8, B is the yellow box, A is the blue box, and C is the red box (Figure 9). And the formulas for GIoU are shown in Equations (3) and (4):

$$GIoU = IoU - \frac{|C - (A \cup B)|}{|C|} \tag{3}$$

$$GIoU\_loss = 1 - GIoU = 1 - (IoU - \frac{|C - (A \cup B)|}{|C|}) \tag{4}$$

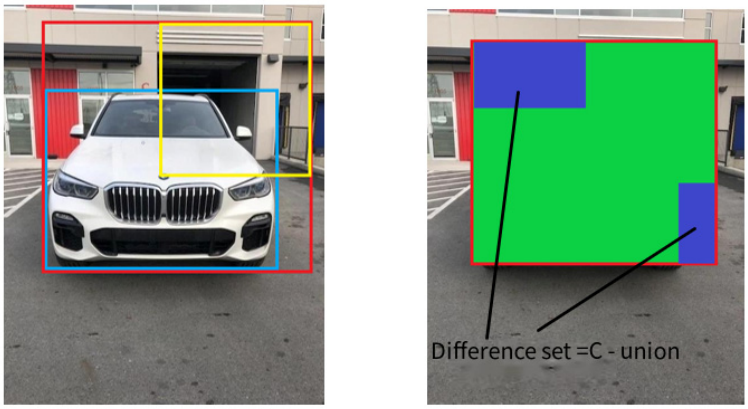


Figure 9. Schematic diagram of GIoU calculation.

GIoU introduced the test box C, which consisted of the combination of the yellow prediction box B and the blue actual box A. The GIoU calculation diagram is shown in Figure 9. In addition to considering the relationship between the prediction box and the actual box, GIoU also introduces the test box. Nevertheless, GIoU cannot play the actual effect while the two boxes are in the horizontal state. Here, the CIoU [31] function is substituted by the GIoU function, and the loss function equation for DIoU [32] is given in Equation (5):

$$DIoU\_Loss = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} \tag{5}$$

here, b, b^{gt} denotes the center points of the prediction box and the real box, respectively, ρ represents the distance between the two center points, and c represents the diagonal distance of the minimal closure region that can encompass both the prediction box and the real box. In addition, the impact factor av is introduced, along with the horizontal to vertical ratio.

The improved formulas of GIoU and CIoU_Loss are shown in Equations (6) and (7).

$$CIoU = IoU - \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{6}$$

$$CIoU\_Loss = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \tag{7}$$

The parameter expression representing the penalty in the formula α is shown in Equation (8), and v represents the standard that can measure whether the aspect ratio is consistent, and the expression is shown in Equation (9):

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{8}$$

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

The modified formula demonstrates that the convergence rate of CIoU is substantially faster than that of IoU.

4. Test and Result Analysis

4.1. The Experiment Platform

Ubuntu18.06 is the operating system version of the training experiment machine for the model presented in this paper. Tables 3 and 4 detail the experimental setting and hardware and software configurations.

Table 3. The development environment.

Hardware Name	Version Number
Processor	AMD Ryzen 5 5600X 6-Core Processor (3701 MHz)
Graphics card	NVIDIA GeForce RTX 3060 12G
Memory	16 GB

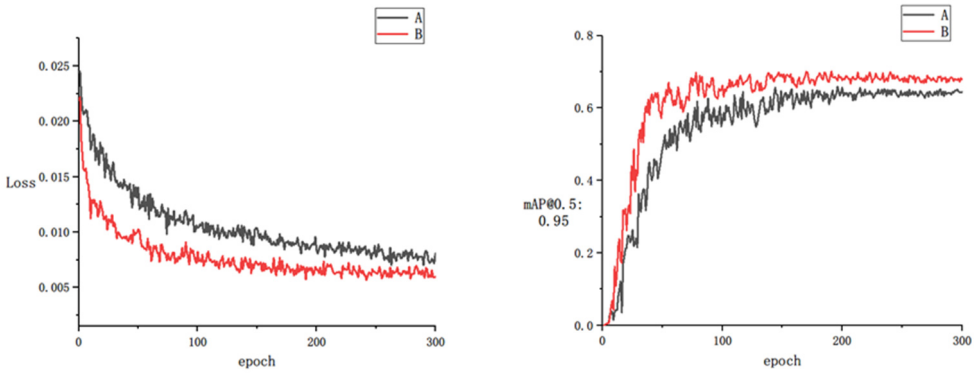
Table 4. Software environment.

The Specific Environment	Version Number
Python	Python3.8
CUDA	11.1
CUDNN	11.3

4.2. Comparison of Training Results

Beginning with an initial learning rate of 0.01, SDG was used to optimize algorithm parameters and the cosine annealing approach was employed to dynamically modify the

learning rate. The weight attenuation coefficient was set to 0.0005, the learning momentum was set to 0.937, and the Batch-Size to 8. In this experiment, 300 epochs were trained to examine the overfitting issue in the training process. Model A is denoted by the black curve, while model B is defined by the red curve. Models A and B are trained together, and the training outcomes are depicted in Figure 10.



(**a**) (**b**)

Figure 10. Comparison of AB model training parameters before and after improvement.

Figure 10a demonstrates that, compared to the loss value before improvement, the improved model exhibits a more pronounced drop and a faster convergence speed. Figure 10b demonstrates that the mAP value of the enhanced model is 3.1% greater than that of the previous model. Overall, the new model is more precise and has a faster convergence rate. The comparison of model performance before and after loss function enhancement is shown in Table 4. As shown in the Table 5, the loss value of model B's parameter was lowered by 2.8% compared to the model before improvement, showing that the convergence speed of the revised model was greatly increased. The mAP% value of the new model was 2.1% greater than previously, and its accuracy was enhanced. Furthermore, both Recall and Precision are greatly enhanced following enhancement. In conclusion, the enhancements to the YOLOv5 model presented in this research greatly increase the performance and convergence speed and precision.

Table 5. Performance comparison between model A and model B.

Model	Loss	mAP@0.5:0.95	Precision (%)	Recall (%)
Model A	0.0068	0.651	91.9	92.5
Model B	0.0070	0.672	93.9	94.5

4.3. Algorithm Improves Visual Contrast

The same image is used to examine the effect difference of the activation function of YOLOv5's algorithm improvement before and after, in order to more intuitively illustrate the improvement of the algorithm's accuracy and speed in image recognition. As shown in Figure 11a depicts the original input image, Figure 11b depicts the accuracy of vehicle recognition before the activation function is enhanced, and Figure 11c depicts the vehicle recognition after the activation function has been enhanced. After an object is discovered, there will be text indicating the sort of object detected, along with a recognition accuracy indicator.

(**a**) (**b**) (**c**)

Figure 11. Visualization comparison before and after activation function improvement.

As shown in the picture above, the confidence level increases when the activation function in YOLOv5's algorithm is enhanced, the accuracy of target recognition is enhanced, and the model's performance is further enhanced.

The diagram depicts the experimental outcomes of adding the CBAM module to the YOLOV5s network architecture. As shown in Figure 12, Figure 12a is the original image, Figure 12b is the detection image before CBAM improvement, and Figure 12c is the detection image after CBAM improvement. The results indicate that the modified CBAM algorithm has significantly enhanced the detection of small objects. Tiny items that were previously undetectable can now be detected, and the confidence level has been increased; nevertheless, the improvement effect on the identification of large objects is not readily apparent. Hence, the algorithm's performance is further enhanced with the addition of the CBAM module.

(**a**) (**b**) (**c**)

Figure 12. Visualization comparison before and after CBAM function improvement.

Figure 13 is a comparison of the experimental outcomes before and after the improvement of the loss function. Figure 13a is the original picture, Figure 13b is the detection graph before the improvement of the loss function, and Figure 13c is the detection graph after the improvement of the loss function. The preceding graph demonstrates that the modified algorithm increases the accuracy of vehicle detection, as well as the convergence speed of the loss function and the recognition speed.

(**a**) (**b**) (**c**)

Figure 13. Visual comparison of loss function before and after improvement.

4.4. Experimental Verification of the Improved Algorithm

To further demonstrate the superiority of the enhanced algorithm, an ablation experiment was undertaken to evaluate the model's performance. The enhanced activation function, attention mechanism and loss function were utilized to validate the model's performance.

Table 6 demonstrates that after the activation function was enhanced the mAP value grew by 2.1%, while Precision and Recall also increased slightly. Adding the attention mechanism to the Backbone network increased the mAP value by 2.9%. mAP value increased by 1.5%, Precision increased by 0.9%, and Recall increased by 1.1% after the loss function was adjusted. After enhancing the three algorithmic components, the algorithm's Precision and Recall are enhanced by 2% and 2%, respectively. In conclusion, based on the ablation experiment conducted after the algorithm improvement, it can be concluded that the improved model performance was significantly enhanced in terms of confidence, precision, and recall compared to that of the previous model, thereby effectively improving the model performance.

Table 6. Comparison table of ablation experiments.

YOLOv5s	Activation Function	Mechanism of Attention	Loss Function	mAP@ 0.5:0.95(%)	P (%)	R (%)
√				65.1	91.9	92.5
√	√			66.5	92.2	92.8
√		√		67.0	93.1	94.1
√			√	66.1	92.8	93.6
√	√	√	√	67.2	93.9	94.5

In addition, YOLOv5s is compared to YOLOv4, YOLOV4-Tiny, and Faster-RCNN, which are typically used to evaluate the performance of each algorithm. Table 7 compares the performance of several algorithms.

Table 7. Performance comparison of different detection methods.

Method	Model Storage Size (MB)	mAP@0.5 (%)	P (%)
Faster-RCNN	186	83.7	83.8
YOLOv4	113.9	93.1	93.3
YOLOv4-tiny	30	83.4	87.3
YOLOv5s	24.5	87.4	91.9
Our approach	24.7	91.5	94.5

As seen in the table above, the YOLOv5s algorithm requires the minimum amount of memory to operate and performs well in terms of confidence level and precision. Following the enhancement of the experimental algorithm, the mAP@0.5 value and accuracy have been further enhanced and the method's overall performance has been enhanced.

In conclusion, the improved algorithm is superior to the previous algorithm in terms of object recognition speed and accuracy, effectively addresses the disadvantage of low accuracy in the recognition of small objects, and improves the shortcomings of the previous algorithm, such as vanishing gradient and low confidence, making the algorithm more practical and efficient.

5. Conclusions

In this paper, the original YOLOv5s algorithm was enhanced in order to address the issues present in the basic YOLOv5s algorithm, including the disappearing model training gradient, tiny target object recognition accuracy and poor convergence speed of loss function. First, the new activation function is substituted for the old model's activation

function, which successfully mitigates the gradient descent of the Leaky ReLU function. Then, to address the issue that the YOLOv5s algorithm has a low recognition rate for small objects, the CBAM module is included to improve the algorithm's feature extraction for small and medium-sized objects. Lastly, the CIoU loss function replaces the original YOLOv5s loss function. The improved detection algorithm proposed in this paper is superior to the YOLOv5s algorithm prior to the improvement in terms of accuracy, mAP, Recall, etc., so the improvement of the algorithm can effectively solve the problems of gradient loss, low accuracy of small object recognition, and slow reasoning speed in the original algorithm, and the improved method has clear benefits.

Author Contributions: Conceptualization, L.S.; Data curation, C.L.; Formal analysis, J.L.; Investigation, H.W.; Methodology, L.S.; Supervision, C.L.; Validation, H.W.; Writing—original draft, H.W.; Writing—review & editing, H.W. and J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The load forecasting data used to support the results of this study has not been provided because it is private data of enterprises.

Conflicts of Interest: The authors declare that there is no conflict of interest regarding the publication of this paper.

References

1. Divakarla, K.P.; Emadi, A.; Razavi, S.A. Cognitive advanced driver assistance systems architecture for autonomous-capable electrified vehicles. *IEEE Trans. Transp. Electrif.* **2019**, *5*, 48–58. [CrossRef]
2. Koustanaï, A.; Cavallo, V.; Delhomme, P.; Mas, A. Simulator training with a forward collision warning system: Effects on driver-system interactions and driver trust. *Hum. Factors* **2012**, *54*, 709–721. [CrossRef] [PubMed]
3. Vapnik, V.; Levin, E.; Cun, Y. Measuring the VC-Dimension of a Learning Machine. *Neural Comput.* **1994**, *6*, 851–876. [CrossRef]
4. Sri, M.S. Object detection and tracking using KLT algorithm. *Int. J. Eng. Dev. Res.* **2019**, *7*, 542–545.
5. Joseph, R.; Santosh, D.; Ross, G.; Ali, F. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
6. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014.
7. Everingham, M.; Gool, L.V.; Williams, C.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–308. [CrossRef]
8. Hamsa, S.; Panthakkan, A.; Al Mansoori, S.; Alahamed, H. Automatic Vehicle Detection from Aerial Images using Cascaded Support Vector Machine and Gaussian Mixture Model. In Proceedings of the 2018 International Conference on Signal Processing and Information Security (ICSPIS), Dubai, United Arab Emirates, 7–8 November 2018; pp. 1–4.
9. Zhang, H.; Wang, Y.; Dayoub, F.; Sünderhauf, N. VarifocalNet: An IoU-aware Dense Object Detector. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020.
10. Wang, Z.; Jun, L. A review of object detection based on convolutional neural network. In Proceedings of the 2017 36th Chinese Control Conference (CCC), Dalian, China, 26–28 July 2017; pp. 11104–11109.
11. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
12. Han, X.; Zhong, Y.; Cao, L.; Zhang, L. Pre-Trained AlexNet Architecture with Pyramid Pooling and Supervision for High Spatial Resolution Remote Sensing Image Scene Classification. *Remote Sens.* **2017**, *9*, 848. [CrossRef]
13. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
14. Cai, Z.; Vasconcelos, N. Cascade R-CNN: High Quality Object Detection and Instance Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 1483–1498. [CrossRef] [PubMed]
15. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
16. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. R-FCN: Object detection via region-based fully convolutional networks. In Proceedings of the 30th International Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 379–387.
17. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single shot multi-box detector. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.

18. 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.
19. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
20. Zhang, F.; Yang, F.; Li, C. Rapid Vehicle Detection Method Based on Improved YOLOv3. *Comput. Eng. Appl.* **2019**, *2*, 3–8.
21. Wang, F.; Zhang, J.; Lu, G. Vehic YOLOv4: Optimal Speed and Accle Information Detection and Tracking System Based on YOLO. *Ind. Control. Comput.* **2018**, *7*, 89–91.
22. Ding, B.; Yang, Z.; Ding, J.; Liu, J. Highway tunnel stop detection method based on improved YOLOv3. *Comput. Eng. Appl.* **2021**, *23*, 234–239.
23. Fu, J.; Su, Q.; Zhang, D.; Li, J. A Road Multi-Object Detection Method Based on YOLOv3. *Comput. Sci. Appl.* **2021**, *11*, 207–216. [CrossRef]
24. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
25. Jocher, G. Yolov5[EB/OL]. Code Repository. 2020. Available online: https://github.com/ultralytics/yolov5 (accessed on 10 January 2022).
26. Jiang, T.; Cheng, J. Target recognition based on CNN with LeakyReLU and PReLU activation functions. In Proceedings of the 2019 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), Beijing, China, 15–17 August 2019; pp. 718–722.
27. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
28. Liu, Y.; Shao, Z.; Hoffmann, N. Global Attention Mechanism: Retain Information to Enhance Channel-Spatial Interactions. *arXiv* **2021**, arXiv:2112.05561.
29. Jiang, B.; Luo, R.; Mao, J.; Xiao, T.; Jiang, Y. Acquisition of localization confidence for accurate object detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 816–832.
30. He, J.; Erfani, S.; Ma, X.; Bailey, J.; Chi, Y.; Hua, X.S. α-IoU: A Family of Power Intersection over Union Losses for Bounding Box Regression. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 20230–20242.
31. 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]
32. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance IoU Loss: Faster and Better Learning for Bounding Box Regression. In Proceedings of the 2020 Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), New York, NY, USA, 7–12 February 2020; pp. 1–8.