Vehicle detection and tracking algorithm based on improved feature extraction

Xiaole Ge[1], Feng Zhou[1*], Shuaiting Chen[1], Gan Gao[1], and Rugang Wang[1]
[1] School of Information Engineering, Yancheng Institute of Technology,
Yancheng, Jiangsu 224051, China
[e-mail: zfycit@ycit.edu.cn]
*Corresponding author: Feng Zhou

Received May 15, 2024; revised July 3, 2024; revised August 7, 2024;
accepted August 26, 2024; published September 30, 2024

Abstract

In the process of modern traffic management, information technology has become an important part of intelligent traffic governance. Real-time monitoring can accurately and effectively track and record vehicles, which is of great significance to modern urban traffic management. Existing tracking algorithms are affected by the environment, viewpoint, etc., and often have problems such as false detection, imprecise anchor boxes, and ID switch. Based on the YOLOv5 algorithm, we improve the loss function, propose a new feature extraction module to obtain the receptive field at different scales, and do adaptive fusion with the SGE attention mechanism, so that it can effectively suppress the noise information during feature extraction. The trained model improves the mAP value by 5.7% on the public dataset UA-DETRAC without increasing the amount of calculations. Meanwhile, for vehicle feature recognition, we adaptively adjust the network structure of the DeepSort tracking algorithm. Finally, we tested the tracking algorithm on the public dataset and in a realistic scenario. The results show that the improved algorithm has an increase in the values of MOTA and MT etc., which generally improves the reliability of vehicle tracking.

Keywords: Vehicle detection, Multi-vehicle Tracking, YOLOv5, Intelligent transportation, Deep learning

1. Introduction

The development of transportation has brought convenience to people's travel. Monitoring the size and direction of highway traffic flow is an important part of modern urban construction and planning control [1]. However, with the increasing socio-economic and technological growth, the size of road traffic has become progressively larger. Problems such as traffic congestion on complex roadways and traffic violations on main roads are potential hazards for traffic accidents. Nowadays, real-time detection and tracking in traffic scenarios is still a hot issue that has been widely discussed and involves many fields such as image processing, computer vision, and artificial intelligence. Vision-based image processing techniques usually include detection and tracking tasks with additional functions such as trajectory extraction and evaluation of traffic information. The collection of informatization, intelligence, and socialization of such algorithmic systems constitutes the Intelligent Transportation System (ITS) [2]. How to make this system more efficient, robustly regulate and rationally allocate resources is a central issue in modern transportation.

Continuous and stable tracking of targets in traffic scenarios is an important research direction in ITS [3]. This field has long been in the spotlight, with multi-faceted application capabilities in automated driving, accident prediction, traffic assessment, etc. [4-5] Unlike traditional radar and laser ranging technologies or vehicle tracking based on big data processing [6], vision-based object tracking has the advantages of low cost, real-time and high performance, and is one of the key technologies in ITS systems. It can be roughly divided into two parts: detection and tracking.

The task of vehicle detection is to find the vehicle target in the video frames and give classification and localization. Traditional detection methods such as "Background Subtraction (BS) + Support Vector Machines (SVM)", "BS + KNearest Neighbor (KNN) "[7], rely on human-specified feature vectors to represent salient features in an image. David and Athira [8] used filters to obtain vehicle features, which were then fed into the SVM to identify vehicles in the image. Yan [9] et al. screened vehicle contours by vehicle shading and used histogram of oriented gradients (HOG) to extract features, but the detection was poor when vehicles were occluded. These methods require specialized knowledge, while not intelligent enough.

The vehicle tracking task is to detect the vehicle target in a video frame and associate it with the same target in the previous frame to be matched by a correlation technique. There are various methods, such as Frame Difference Method for detecting moving objects, Background Subtraction Algorithm, and Optical Flow for detecting pixel changes. These methods achieved continuous tracking based on single-frame images. But it is slow in continuous video frames.

Due to various problems in traffic scenarios, such as scene complexity, camera angle, object occlusion, etc., current object detection and multi-object tracking (MOT) algorithms still face many challenges. This paper is based on the contemporary problems of low accuracy for vehicle detection and poor stability for vehicle tracking. The contributions are as follows:

1) Improvement of the detector: a more detailed multi-scale feature extraction module is proposed, and the attention mechanism is properly integrated to realize more fine-grained feature extraction. It makes the model pay more attention to the vehicle features, and the ability to discriminate vehicles of different sizes is improved, with less background interference. This enhanced accuracy in recognizing vehicles. Combined with the improvement of the box loss function, the anchor boxes are more stable with fewer false detections, and there is no big change in the amount of calculation.

2) Improve the tracker: modify the input feature map according to the proportion of vehicles, and rearrange the vehicle re-identification network module to decrease the redundancy of the neural network in the original algorithm. It makes the matching of vehicle information more accurate. In effect, it enables more accurate tracking of vehicles and reduces the number of ID switch.

Combined with improved detection and tracking algorithms, it solves the problem of detection accuracy as well as ID switch to a certain extent and achieves better relevant indexes on vehicle tracking.

2. Related work

Nowadays, deep learning-based detection methods are becoming more and more popular. From LeNet to ResNet to MobileNet and so on [10-12], neural networks have been revolutionized several times, and classical networks have been widely used in the field of vision. So far, the proposed methods mainly contain two categories: 1) two-stage object detection algorithms represented by Region-based Convolutional Network (RCNN) [13], 2) one-stage object detection algorithms represented by You Only Look Once (YOLO) [14], Single Shot MultiBox Detector (SSD) [15], and so on. The YOLO algorithm is capable of determining and localizing the object class through a single regression. Therefore, the one-stage object detection algorithm is generally better than the two-stage algorithm in terms of speed, but accordingly, the accuracy and stability of the algorithm is a shortcoming.

Accordingly, most of the research in this area has been focused on the fast and high accuracy of the algorithmic models. Zakria [16] et al. achieved better results in remote sensing image recognition by improving YOLOv4. Zhang [17] et al. designed a new anchor box matching algorithm and proposed a non-maximal suppression method based on positional priority to achieve effective detection of trucks. Chen [18] et al. replaced the backbone network of the SSD with the MobileNet-v2 network and introduced the attention mechanism, successfully constructed a lightweight SSD vehicle detection model. It is important to note that most of the data used for vehicle detection is derived from surveillance images, and a large amount of manually labeled data may be required for application scenarios [19]. At the same time, for the video information of the traffic scene, real-time analysis has high requirements for the detection speed, for this reason, YOLO series algorithm is undoubtedly a better choice.

In the YOLO series, currently available up to the YOLOv9 [20] version, its detection performance in various fields has yet to be tested. YOLOv8 [21] has a lightweight module and high performance, but the algorithm contains more convolutional layers and parameters and is slower to train. YOLOv7 algorithm has more fine-grained discriminative ability, and it has high detection accuracy [22-23], but the detection speed is slower. YOLOv5 [24] is one of the more widely used and deployed versions available. Compared to the previous two, YOLOv5 has a lower detection accuracy, but it has greater advantages in training speed, inference speed, model size, etc., which makes it very suitable for real-time detection and tracking scenarios [25].

In the field of object tracking, many modern methods have been proposed compared to the traditional classical methods. Most of them are intelligent and extensible. Cubeta Caballo [26] et al. and Tran [27] et al. used real-time tracking based on the YOLO model. A stochastic approach was applied by Dey [28] et al. Nowadays, Kalman filtering and particle filtering [29] are also common in various algorithms. Song [30] et al. proposed a joint vehicle tracking algorithm based on RSU selection. However, these algorithms have some shortcomings in terms of accuracy or speed. In addition, it can be classified into Detection-Based Tracking

(DBT), Detection-Free Tracking (DFT), and Joint-Detection Tracking (JDT), which fuses the detection features, according to the initialization method or the detection processing method [31]. DFT mode requires manual labeling of the tracked object in the initial frame, and the algorithm is not flexible enough. While JDT mode needs to fuse various features of detection with network generation is more difficult. The DBT has some dependence on the detector, but the application aspects are wider.

SORT (Simple Online and Realtime Tracking) [32], as a detection-based tracking algorithm, achieves simple and efficient tracking by evaluating the correlation between detection and tracking results in terms of IoU through Kalman filtering and matching algorithms. DeepSort [33] is an improved version of SORT, which adds ResNet to extract appearance feature vectors, adds cosine distance to affect the IoU cost matrix, and better solves the ID switching problem caused by occlusion in SORT algorithm. It is suitable to be applied in the scene where a large number of objects appear.

For the partial methods proposed by the people mentioned above, we summarize them in **Table 1**. Currently, the detection and tracking of vehicles have always been a challenging task in computer vision, and its difficulties include but are not limited to, the interference of lighting, weather changes on detection, and large positional gaps between object boxes. All these problems can cause the re-identification process to fail. Therefore, an accurate detector with robustness and a stable and fast tracker is important for tracking tasks in traffic scenarios. Most research nowadays is also done in terms of accuracy and speed or is targeted to achieve specific functionalities.

Table 1. Summary of mentioned methods

Author	Type	specificities	limitations
Zakria et al. [16]	detection	Improved problem of anchor box allocation	The speed still needs to be improved in real-time
Zhang. Y et al. [17]	detection	Propose a new anchor box matching and non-maximal suppression method to promote the recall ratio	Committed to improving recall, with little consideration of precision and classification
Chen. et al [18]	detection	Applying MobileNet v2 to Reduce Network Computations	The accuracy is not optimistic compared to new methods
Cubeta Caballo et al.[26]	tracking	Added count function to the SORT algorithm which imports the detection model	No consideration of weather, occlusion, etc.
Tran et al. [27]	tracking	The tracking method was deployed on multiple recording devices	Brightness as well as reflecting backgrounds may cause some detection problem
Dey et al. [28]	tracking	Develop a stochastic method useful in times of data shortage	Different time and modelling strategies could influence the result
Song et al. [30]	tracking	Develop road side unit (RSU) that enhances tracking result	samples must be shared between RSUs could impose a burden on the network

3. Vehicle Detection and Tracking Methods

3.1 System model

The system uses YOLOv5 as a detector and DeepSort as a tracker to complete the vehicle tracking, its brief process is shown in **Fig. 1**. The specific implementation is to find out the vehicle objects in the video frames possible by YOLOv5 with the position information. After that, it is sent to DeepSort for subsequent trajectory prediction and information matching. The predicted trajectory will attempt to match the detection box in the future. Then, the algorithm assigns the same ID to the vehicles that are successfully matched. A vehicle whose information has been matched will in turn update the trajectory information of it. Keeping the ID of the same target unchanged in continuous frames is a good tracking of the vehicle.

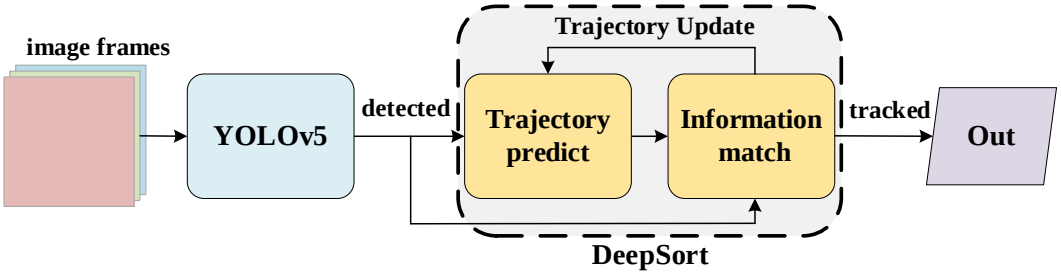


Fig. 1. Flowchart of the system

The entire network structure of YOLOv5s can be roughly divided into four parts: Input, Backbone, Neck, and Head. The input side includes operations such as data enhancement, adaptive boxes, and image sizing. The backbone network extracts the features of the image from multiple channels through deep convolution, residual joining, etc. Finally, the feature map output is unified in size through spatial pyramid pooling. While increasing the number of channels, it facilitates the extraction of features and reduces overfitting. The neck network uses the structure of Feature Pyramid Network (FPN) + Pixel Aggregation Network (PAN) to fuse the information of different-sized feature maps in the backbone network during the sampling process in order to further enhance the detection effect. Finally, it is fed into the head, which is the detector of the model that gives the position of the object of interest in the feature maps of different sizes and predicts its class to complete the detection. Its models are categorized into five types, n, s, m, l, and x, based on size, as shown in **Table 2**.

Table 2. Different models of YOLOv5[24]

Models	Parameters	
	Depth	Width
YOLOv5n	0.33	0.25
YOLOv5s	0.33	0.50
YOLOv5m	0.67	0.75
YOLOv5l	1.00	1.00
YOLOv5x	1.33	1.25

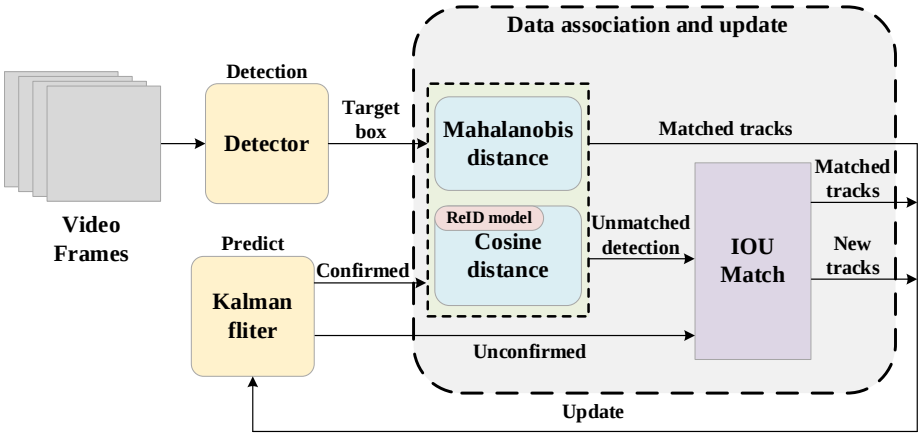


Fig. 2. Flowchart of DeepSort

DeepSort represents the state of an incoming detection object as an 8-dimensional vector $I = (x, y, \gamma, n, \dot{x}, \dot{y}, \dot{\gamma}, \dot{n})$, recording the object's center point coordinates, aspect ratio, picture height, and motion speed corresponding to these four pieces of information, respectively. The Kalman filtering will be used to predict the trajectory of the next possible movement. When the predicted object box and the box of this frame are less than a distance threshold, the two are matched to complete the tracking of the object.

In addition, a simple Convolutional Neural Networks (CNN) is incorporated in DeepSort to extract the appearance features of the objects in the detected frame, which is called the ReID network. After each frame tracking, the extraction of object appearance features is performed once and saved. At each step of execution, a similarity calculation between the appearance features of the detected object in the current frame and the saved features is performed. This similarity will be used as an important discriminative basis during association matching. The flowchart of DeepSort is shown in **Fig. 2**.

3.2 Improved model

3.2.1 Loss function

The 7.0 version of YOLOv5 uses CIoU Loss as the loss function of bounding box by default but does not consider the region between the predicted box and the real box. WIoU [34] weights the IoU by considering the region between the predicted box and the real box, which solves the problem of bias that the traditional IoU may have when evaluating the results, and to some extent, makes the anchor box more accurate and the size more adaptable. The formula is as follows:

$$WIoU = \frac{\sum_{i=1}^{n} w_i IoU(b_i, g_i)}{\sum_{i=1}^{n} w_i}. \tag{1}$$

In Eq. (1), n denotes the number of prediction boxes, b_i denotes the coordinates of the ith prediction box, $g_i$denotes the coordinates of the ith labeled box, $IoU(b_i, g_i)$ denotes the IoU value between the ith prediction box and the labeled box, and w_i is the weight value.

In WIoU, the weight value of each prediction box depends on how much it overlaps with the real labeled box. Prediction frames with greater overlap are given higher weights, and those

with less overlap are given lower weights. Replacing the CIoU with the WIoU allows for a better evaluation of the detection results and gives a more accurate evaluation in the presence of imbalance between large and small objects.

3.2.2 Adaptive attention fusion based on multi-scale features

Convolutional neural networks operate by complex convolution of images so that sub-features are usually distributed across channels and represent various semantic information. Most attention mechanisms are dedicated to focusing on the focal information of the image from either semantic or spatial point of view, such as SE [35], CBAM [36], etc., which are commonly used. This type of attention needs to receive a specified number of channels and divides the channels in the same way when it operates. However, this could cause background noise to affect the global information. SGE (Spatial Group-wise Enhancement) was proposed by Li [37] et al. as a lightweight attention mechanism. The authors propose feature map grouping and give a global vector to filter out the more useful and featured channel information when grouping, while excluding the interference of noise to some extent. Its structure is shown in **Fig. 3**.

First, the input feature maps are divided into G groups by channel, and average pooling is done for the sub-feature maps of each feature map. The global vector $\boldsymbol{g}$ is obtained after this and is used to determine the similarity between the sub-features and the global features in terms of vectors:

$$\boldsymbol{g} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x_i}. \tag{2}$$

in Eq. (2), $m = H \times W$, $\boldsymbol{x_i}$ is the vector of subfeature maps in the group. Then, a dot product is performed with the atomic feature map to obtain the attention at each position:

$$c_i = \boldsymbol{g} \cdot \boldsymbol{x_i}. \tag{3}$$

This is followed by a BN layer and Sigmoid function to get $\sigma(a_i)$. Dot product this with the original inputs to obtain the output vector $\hat{\boldsymbol{x}}_{\boldsymbol{i}}$ of the group:

$$\hat{\boldsymbol{x}}_{\boldsymbol{i}} = \boldsymbol{x_i} \cdot \sigma(a_i). \tag{4}$$

For correct, useful information that is reflected on the vectors as directions more similar to the global vectors, their mutual dot product results in larger values. In this way, useless information such as inter-channel noise can be learned and suppressed in a targeted manner, while at the same time improving the spatial regional accuracy of the features. In addition, SGE does not have complex residuals and lots of convolutions, the number of parameters introduced is very small, which ensures the light weight feature.

In the backbone network of YOLOv5, the C3 feature extraction module uses multiple BottleNeck residuals to propagate gradient information based on the input feature map dimensions. However, this feature extraction structure causes the network to use a large amount of duplicate gradient information when updating parameters, the redundant gradient information being passed in duplicate is not conducive to the learning and training of the network.

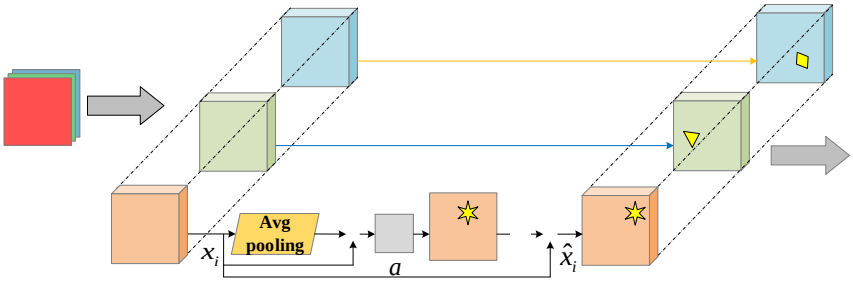


Fig. 3. SGE structure

Nowadays, multi-scale features have been widely used in computer vision tasks and achieved relevant results. Gao [38] et al. proposed a new multi-scale feature residual structure Res2Net, which constructs hierarchical connections within a residual block, and realizes multi-scale feature extraction by group convolution, inter-group fusion, and finally splicing the feature maps with different receptive fields.

Based on the defects of the C3 module and the strong feature extraction capability of Res2Net, the C3_Res2A module is proposed. Abandon the original algorithm's information transfer method, instead, the input n channels of feature map are divided into s groups according to the channels, and each group has its own small convolutional layer corresponding to the n/s output channels. Groups are convolved with each other and passed to the next group, fusing features with each other and doing the convolution operation for the group again. In this way, after layers of convolution, the s larger set of feature maps has a larger receptive field. Finally, the feature maps of different receptive fields are spliced to realize feature extraction in different scales, and this feature extraction method can effectively improve the expressiveness of different features. The formula is as follows:

$$y_i = \begin{cases} x_i, i = 1 \\ K_i(x_i + y_{i-1}), i \in (1, s] \end{cases}. \tag{5}$$

In Eq. (5), the s is the number of groups, and x_i denotes the number of i group's feature map, K_i denotes the convolution kernel corresponding to the group i in exception of group one, and y_i is the output of x_i after the convolution. In C3_Res2A the s is set to 4, the convolutional layer of each group adopts Convolution-BN-SiLU operation, and the size of the convolution kernel is 3, so that it has a better effect when splicing the dimensions in the concat layer. After the residual output, the SiLU activation function is used to ensure the validity of the feature maps and facilitate the convergence and accuracy enhancement during training. Finally, by integrating the SGE attention mechanism, the feature map is divided again into 8 groups, which reduces the inter-channel noise generated by the grouping in the previous step. This likewise reduces the model's attention to the background. Overall, the ability of the neural network to recognize features is greatly enhanced.

The overall C3_Res2A module presents a grouping-splicing-grouping-splicing process, as shown in **Fig. 4**. By replacing the last three C3 layers in the backbone network, the detailed feature extraction of feature maps with different sizes is accomplished, and the key features of the original images are retained to a greater extent. It also reduces inter-channel noise and offers more flexibility in the dissemination of information. In addition, benefiting from the lightweight calculation of SGE, the structure of the algorithm does not increase the number of parameters additionally after the improvement, which does not have a large influence on the detection speed.

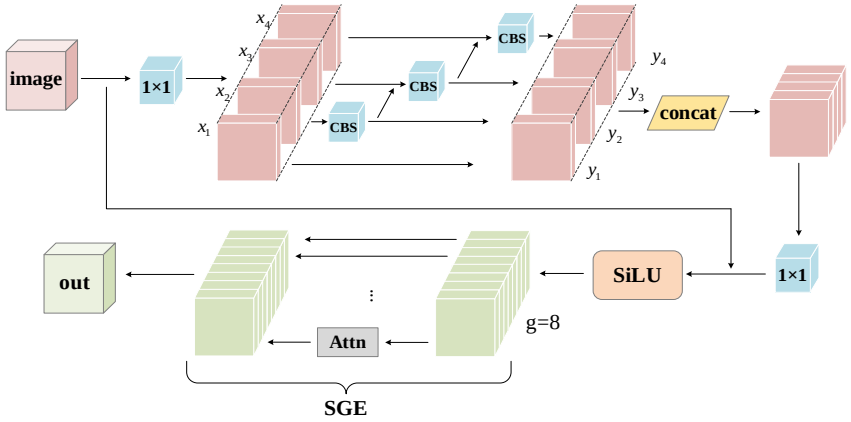


Fig. 4. C3_Res2A structure

3.2.3 Multi-head Self Attention

Traditional CNN-based object detection models are limited by the localization of convolutional operations, which restricts their ability to capture global information in general, the Query(q), Key(k), and Value(v) vectors can be created, and the attention score is obtained by calculating $\boldsymbol{q} \cdot \boldsymbol{k}$, and adjusted to realize the self-attention mechanism by multiplying with the weights vector $\boldsymbol{v}$**.** Which is shown as follow:

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{6}$$

Multi-Headed Self-Attention (MHSA) accompanied by Vision Transformer (ViT) [39] has gained attention. It through multiple matrices W_n Learning Q,K,V weights of matrix vectors to get weights of multiple attention heads. Multiple focuses of attention are obtained globally, and feature recognition with semantic association information is learned from multiple perspectives, enabling the model to extract interrelated information from a large domain. This helps in better learning of weight assignments in scenarios where a large number of objects are occurring in the model.

Q,K,V is a matrix form of $\boldsymbol{q},\boldsymbol{k},\boldsymbol{v}$ vector, and d_k is the dimension of the vector $\boldsymbol{k}$. By fully connected dimension lifting, each header does self-attention to get Z_i , use the fully connected layer to splice the result of each head, and then get Z^0, when outputting the corresponding feature information. Finally, multiply the weight matrix W^0, By evaluating feature information from multiple perspectives, we get the results. The structure is shown in **Fig. 5**.

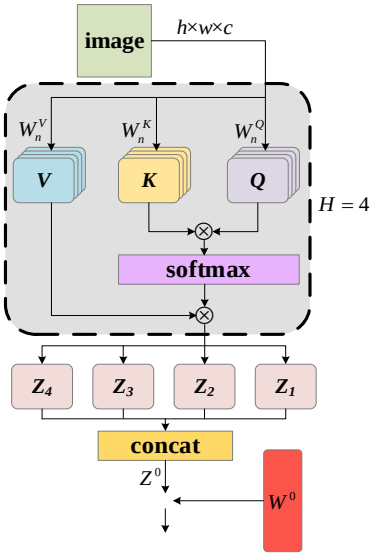


Fig. 5. Structure of MHSA

The overall structure of the improved YOLOv5 is shown in **Fig. 6**. In the backbone, C3 layers are improved by replacing the last three C3 layers with C3_Res2A modules. The size of the feature map at its location is just about the same as that predicted by the detection head. Doing so enables the original image feature information to be preserved when the three-size feature maps are spliced with the post-network layers. The remaining C3 layers in the backbone and neck are replaced with C3_Res2 modules. This module is a multiscale feature extraction module that does not integrate SGE, and its purpose is to enhance feature extraction without bringing in too many parameters, while preventing the model from overfitting. Then, going through the SPPF layer makes the network maximize the number of channels, at which time the semantic information is the richest. After neck network adds MHSA layer to do self-attention mechanism for a large amount of semantic information and achieve differential segmentation of the feature maps through multiple heads.

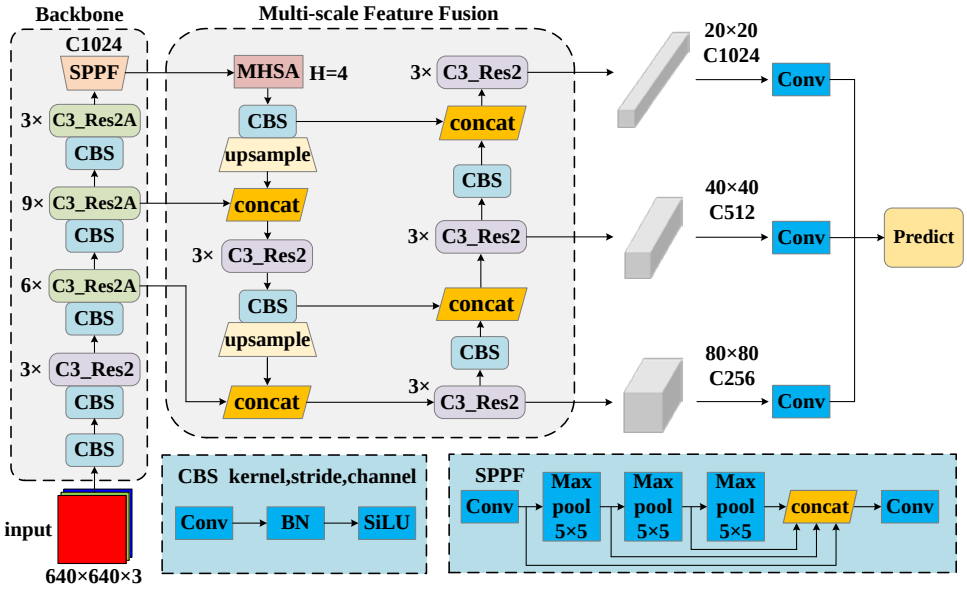


Fig. 6. The structure of improved YOLOv5

3.2.4 Adaptive adjustment of Reid network

In DeepSort algorithm, ReID network gets the object from anchor box in the way of matting image, extracts the features, and saves them. Every time the step is executed a similarity match is performed with the appearance features stored in the previous frame, giving the object ID based on the data association. To adapt it to track vehicle targets, its ReID network was adapted as follows:

1) Resize the inputs and the sizes in each network layer to make the feature map more consistent with the vehicle size characteristics. This enables complete and non-redundant extraction of features in the ReID network when capturing a single frame image.

2) The second convolutional layer in the original algorithm is canceled to reduce the extra parameters brought by it. Add two new sets of residual convolutional layers Residual 9-10 to delineate more vehicle feature semantics, making the re-identification network more detailed and precise for feature extraction.

3) The fully connected layer before the output of the original network structure is modified to an average pooling layer. After suitable adjustment of the network parameters, the same output dimension as the original algorithm can be achieved but the feature information is better. The improved ReID network is shown in **Fig. 7**.

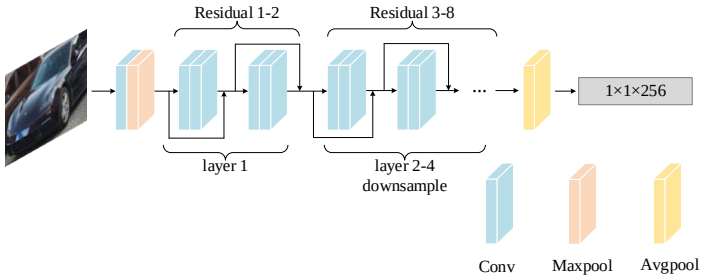


Fig. 7. Structural diagram of the improved ReID network

4. Experiments and analysis of results

4.1 Experimental preparation

The dataset used in the experiment is UA-DETRAC [40] dataset, The dataset covers a variety of traffic scenarios such as highways, road intersections, etc., and the vehicle classes are categorized into cars and other social vehicles, which is designed to train vehicle detection and vehicle tracking. Since most of the images in the huge dataset are made up of continuous frames, the image features of the neighboring frames are almost the same. Therefore, taking one image at an interval of 10 frames from the original dataset will reduce the number of samples and make it easier to train as well as prevent the model from overfitting.

In DeepSort, the VeRI [41] dataset is used for ReID training. This is a large-scale benchmark dataset established in real-world monitoring scenarios for vehicle Reid training that contains more than 50,000 images of vehicles. In this paper, about 3,000 images of different vehicles are selected for training, ranging from about 50 images for each vehicle. About 20% of them are randomly used as test sets.

The YOLOv5 used in this paper is version 7.0. On the input side, the data will be preprocessed, mainly by Mosaic data enhancement and anchor box adjustment. For the input images, it will take random cropping, splicing rotation, and other operations. For the predicted anchor box, the adaptive strategy solves the problem that the aspect ratio of different objects varies.

The hardware platform used for the experiment is listed as follows: CPU: 15 vCPU Intel(R) Xeon(R) Platinum 8358P CPU @ 2.60GHz, GPU: NVIDIA RTX 3090, and others. The software platform uses Ubuntu 20.04 as the operating system, the deep learning framework is Pytorch version 1.11.0, the acceleration environment is Cuda 11.3, etc.

4.2 Results of vehicle detection

The training epoch is set to 300, starting from the original algorithm and adding one improvement module per experiment. After many rounds of iterations, all of them can reach convergence, and the best weights in the training process are used as the training results. Using the trained weights to test the dataset and comparing with each other, the ablation experiments for improving YOLOv5 are as follows:

Table 3. Improved algorithm ablation experiments

YOLOv5	WIoU	C3_Res2	MHSA	SGE	mAP.5	mAP.5-.95	GFLOPs
✓					0.703	0.518	15.8
✓	✓				0.723	0.530	15.8
✓	✓	✓			0.747	0.553	14.4
✓	✓	✓	✓		0.754	0.553	15.1
✓	✓	✓	✓	✓	**0.760**	**0.554**	15.1

As can be seen from **Table 3**, as improvements are applied to the various modules of the network, both the mAP.5 and mAP.5-.95 values gradually increase, but the Giga Floating point Operations Per Second (GFLOPs) decreases rather than increases. There is a large improvement in accuracy with the addition of the C3_Res2 module, indicating that this module has a strong performance for multi-scale extraction and fusion of features. Finally, it is adaptively fused with the attention mechanism, the SGE module enables the model to reach a higher level of detection accuracy. Experiments show that the improved algorithm improves the mAP.5 value by 5.7% and reduces the floating-point operations by 0.7GFLOPs, which achieves high accuracy with low calculation.

Fig. 8 shows the comparison between the improved model and the original yolov5 model. The left side is the input image and the middle and right side are the visualization images of the corresponding model. This is a visualization of a multi-channel image that has gone through a feature extraction layer in a neural network, after which the channels are merged to take out the image as can be seen from the figure, there is a clear difference between the two regarding the effectiveness of feature extraction. The improved model can classify the objects in the image in a more detailed way and retain the object features more completely, which is more conducive to the classification of the object. Meanwhile, the improved model processed images with less noise information, making the detection algorithm more focused on the target vehicle rather than irrelevant background or noise information. In the case of complex road scenes, the detector is easy to be affected by environmental factors, such as roadside vegetation, flower beds and buildings, etc., so our improved model has a better ability to detect vehicle targets in this situation.

Moreover, we also compared with some popular attention mechanisms today, replacing the SGE module at the same location with these attention mechanisms to demonstrate the performance of the SGE module. **Table 4** shows the specific experimental data.

Table 4. Comparison of different attention mechanisms

Method	mAP.5	mAP.5-.95	GFLOPs	Parameters/M
YOLOv5 [24]	0.703	0.518	15.8	**7.02**
with SimAM [42]	0.742	0.531	**15.1**	7.29
with SE [35]	0.745	0.538	**15.1**	7.33
with CBAM [36]	0.746	0.543	**15.1**	7.33
ours	**0.760**	**0.554**	**15.1**	7.28

We chose two classical attention mechanisms, SE and CBAM, and the newly proposed SimAM attention mechanism for comparative experiments. The SE attention mechanism is the classical channel attention, which enhances the effective information of an image by learning the important differences between channels. Although the global features are effectively extracted, the lack of consideration of spatial dimensions leads to insufficient attention to key features. CBAM combines both channel and spatial dimensions while learning the importance of their respective dimensions, but its computational cost is high due to considering both dimensions, so it is slow to train and infer. SimAM introduces a similarity attention mechanism for capturing similarity information through an energy function. Its number of parameters is small and flexible to use. However, it does not perform well on the present dataset. The SGE attention mechanism has a better effect while being relatively lightweight. It has better play in transportation scenarios with limited arithmetic resources.

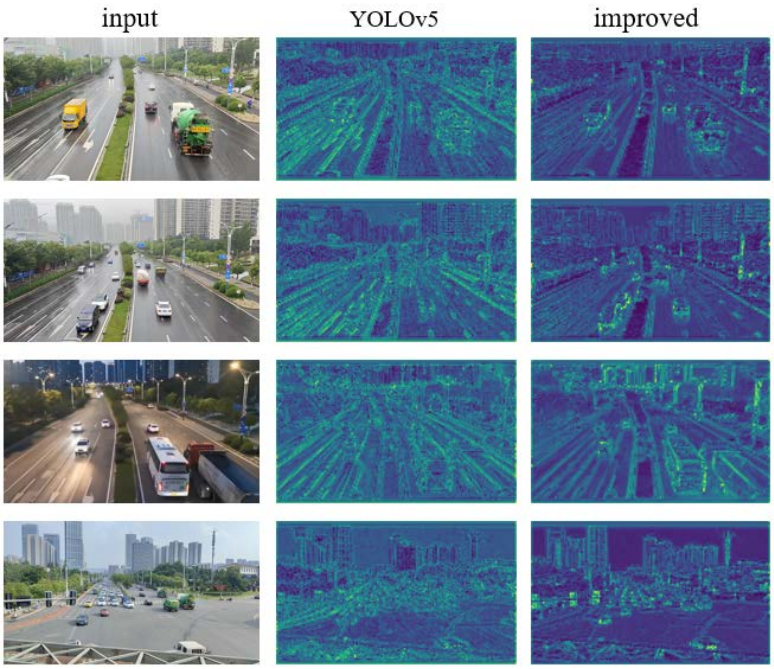
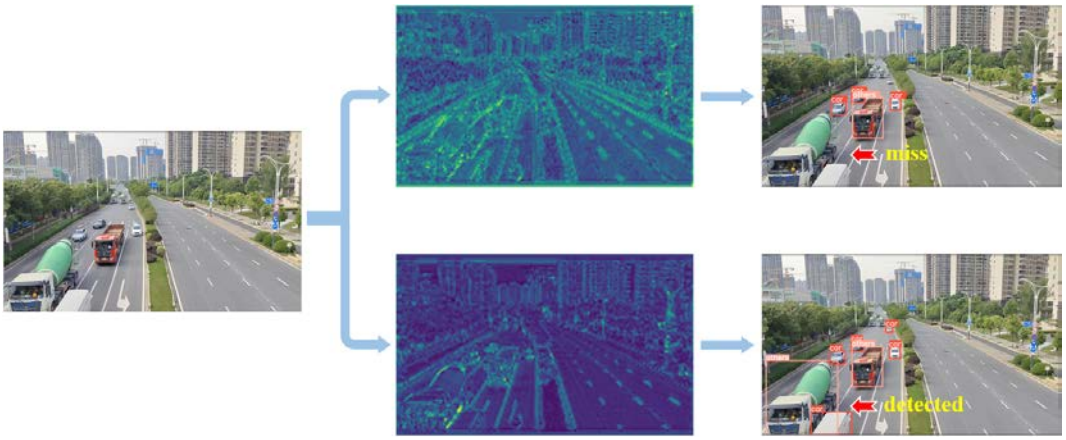


Fig. 8. Comparison of feature extraction effect

The detection results of the YOLOv5 model and the improved model are presented partly in **Fig. 9**. **Fig. 9 (a)(b)(c)** show the real traffic scenes we captured, the COCO dataset, and the

road vehicle images in UA-DETRAC, respectively. Located on the left side of the group figure are the original input images, the middle and right sides are the visualization and output images. Where the top side is the original model output effect and the bottom side is the improved effect. In **Fig. 9 (a)**, the original model fails to detect the cement tanker, probably due to the lack of learning about large vehicles such as it. The improved model is able to do detailed segmentation of the object in the lower left corner of the image through multi-scale feature extraction. It is then possible to classify the cement tanker into the category of others and can detect only a portion of the cars that appear next to this vehicle. In **Fig. 9 (b)**, the original model incorrectly detects the background target, and the anchor box size is similarly unreasonable for the delivery vehicle on the right side. It can also be seen from the feature extraction map that the vehicle on the right side blends in with the background. The feature map extracted by the improved model reduces the background noise and no false detection occurs. Also, thanks to the addition of WIoU, the assigned weights for anchor box localization are improved, and in this way, the anchor box size of the right vehicle is optimized. In **Fig. 9 (c)**, the original model misidentifies the road sign as a car. For the car at the bottom of the picture, the given anchor box is also deformed and does not contain the object completely. It can be seen that the original model does not learn enough, while the improved model solves the above problems.

In the field of object detection, the commonly used and effective methods are Faster-RCNN, SSD, and so on. To verify the performance of the improved algorithm more reliably, this paper compares with the mainstream methods and some of their improved algorithms, the results are shown in **Table 5**. It can be seen that compared to other algorithms; our algorithm has higher accuracy. In general, the mAP.5 value is a widely accepted index for consideration. In the experiments, the index of this paper's algorithm surpasses all the algorithms in relative comparison. Compared with the original YOLOv5 algorithm, and the traditional Faster-RCNN and SSD algorithms, our algorithm has a significant advantage in detection accuracy. For high-precision anchor box requirements, that is, mAP.5-.95, our algorithm does not have an obvious advantage. However, such indexes are generally used for detection tasks with high-precision requirements, and YOLO-FA and YOLOV8s, which have advantages in this index, have too large arithmetic requirements and are more dependent on devices. The YOLOv8n and YOLOv9t, which have the advantage concerning the computation, have a lower number of parameters, but at the sacrifice of accuracy. Our algorithm in this paper considers a lighter calculation, combined with a higher mAP.5 accuracy, it can be better adapted to the task of traffic detection in general.



(a)

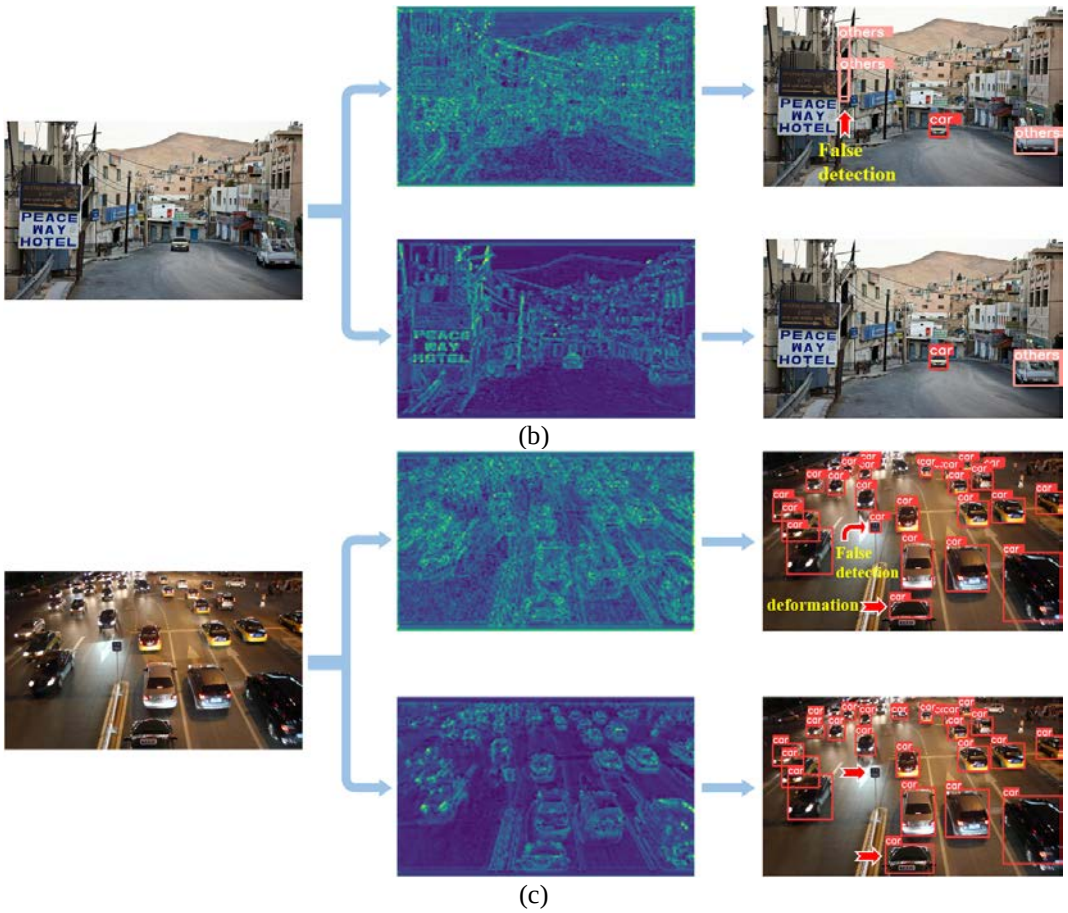


Fig. 9. Comparison of detection effect

Table 5. Comparison of different object detection algorithms

Method	Input	mAP.5	mAP.5-.95	GFLOPs
Faster-RCNN [43]	640×640	0.663	0.403	60.5
SSD [15]	300×300	0.684	-	61.1
YOLOv4 [44]	416×416	0.657	0.473	59.7
CPAM [45]	640×640	0.706	-	-
with Transformer [46]	640×640	0.697	0.549	-
DV3_IBi_YOLOv5s [47]	640×640	0.712	-	-
YOLO-FA [48]	640×640	0.700	**0.598**	26.7
YOLOv5s [24]	640×640	0.703	0.518	15.8
YOLOv8n [21]	640×640	0.739	0.549	**8.1**
YOLOv8s [21]	640×640	0.758	0.577	28.4
YOLOv9t [20]	640×640	0.750	0.514	10.7
Ours	640×640	**0.760**	0.554	15.1

4.3 Results of vehicle tracking

Embed the trained and improved YOLOv5 algorithm into DeepSort as a detector in object tracking. Similarly, DeepSort's ReID network uses the trained network weights. We test the vehicle tracking effect on the UA-DETRAC dataset and filmed scenes, aiming to verify the

overall accuracy and stability of the tracking. The MOTA metric provides a comprehensive measure of the algorithm's accuracy, and its exact calculation is given by (7):

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t}. \quad (7)$$

In the Eq. (7), FN_t is the number of missed detections, FP_t is the number of false detections, $IDSW_t$ is the number of ID switch for a target, and GT_t is the correct labeling. MOTA combines these into one value based on each site of time, providing a reasonable validation method for overall tracking, and is currently the most widely accepted evaluation index. MT is the percentage of time in which all the labeled trajectories can be successfully matched more than 80% of the time; the larger the value of MT, the more trajectories can be successfully matched, and the effect of continuous tracking is more stable. In addition, we also compare other results such as FPS and number of IDs.

For the improved ablation experiments of YOLOv5 with DeepSort, we combined multiple videos from the UA_DETRAC test set to test the corresponding indexes and compare the effects with each other. Second, together with the two videos, we verified several times the change in the number of tracking IDs before and after the improvement. Video1 is the video we took from a road surveillance position and Video2 is the video in the dataset. Method A is the original algorithm; Method B only improves DeepSort's network structure without the improvements of YOLOv5; Method C only improves YOLOv5 and does not adjust DeepSort's ReID network; and, finally, D is the method in which we improve both. The experimental test indexes are MOTA, MT, FPS, and the number of vehicle IDs. The experimental variables are kept constant during the test. The experimental results are shown in **Table 6**:

Table 6. YOLOv5+DeepSort ablation experiments

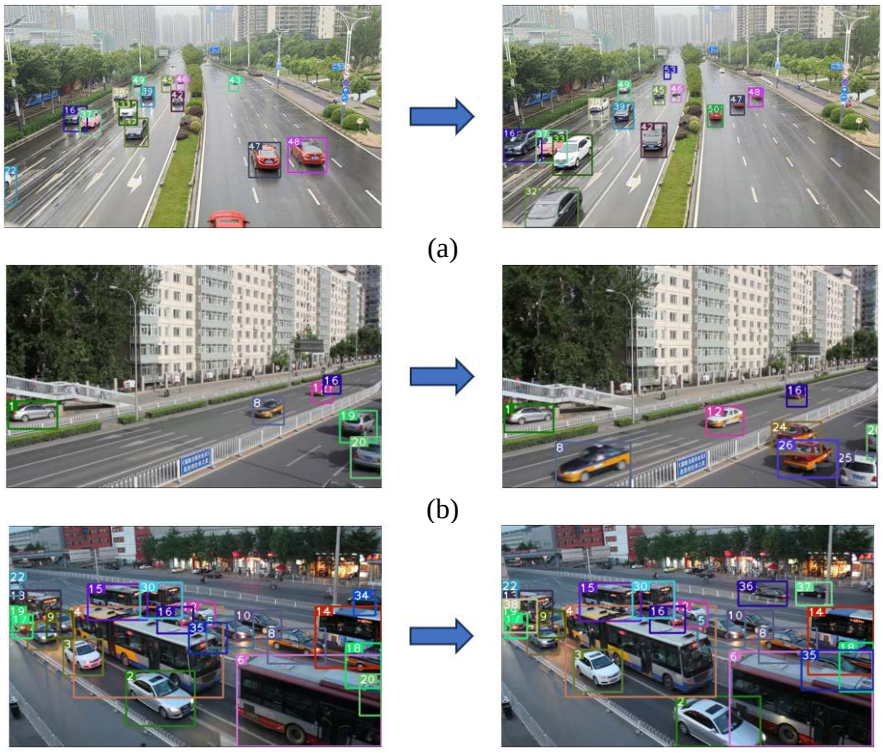
Method	MOTA	MT	FPS	Video1 IDs	Video2 IDs
A	23.1	27.4	30.9	100	160
B	23.9	27.5	30.8	90	130
C	28.8	31.6	31.6	72	128
D(ours)	29.6	32.1	31.4	66	121

From the table, it can be seen that the MOTA and MT values of the C and D methods with improved detectors have a greater improvement compared to the A and B. This proves that the accuracy of the detector has a large influence on the tracking effect, while our method of improved detector benefits the performance of the overall tracking algorithm. Improvements to the detector before the tracker improvements provide a 5.7% increase in the MOTA metric; 4.2% and 4.6% for the MT metrics, respectively. On the other hand, comparing A and B with C and D, we can find that the DeepSort algorithm after improving the ReID network has a small improvement in accuracy and tracking stability, which suggests that our improvement of DeepSort is also beneficial. On top of this effect, the FPS of the algorithm before and after the improvement does not change significantly. Based on the number of IDs in the tracking video, it can be noticed that due to the improvement of the ReID network, the tracker can better recognize the appearance features of the same vehicle after capturing a vehicle target, and is less susceptible to being interfered which could be caused by ID switching during matching information between frames. Finally, the improved algorithm enhances the MOTA and MT values by 6.5% and 4.7%, respectively. This proves that our improved algorithm tracks vehicles more stably and estimates the number more accurately without slowing down the speed. **Fig. 10 (a)(b)(c)** show Video1 and Video2 and other dataset

videos, with their tracking effects.

Fig. 11 (a)(b) shows a comparison of the differences between the original(a) and improved(b) tracking algorithms. We can see that during a small part of the original algorithm's tracking process, the car with ID 2 experienced bus occlusion, and at the end of the occlusion the car's ID switched to 152. In addition, the algorithm detected redundant anchor boxes for the right side car. The improved model could solve this problem. Before the occlusion, the car's ID is 1. After a short period of occlusion by the bus, the algorithm is still able to recognize the car and gives it an ID of 1, which is successfully tracked.

To verify the effectiveness of the proposed vehicle tracking algorithm, we also compared it with the classical or improved methods proposed by others. The dataset is the test set of UA-DETRAC and the results are shown in **Table 7**.



(a) (b) (c)

Fig. 10. Results of tracking effect

As can be seen from the table, most of the results have low MOTA values of just under about 30%. This is due to the fact that the images in the UA-DETRAC dataset contain different weather and complex traffic backgrounds. Our method proposed in this paper has better results in MOTA values, not the highest but close to the highest value of 30.3%. This shows that our algorithm performs well in the problems of tracking accuracy, and ID switching. In terms of MT values, our proposed method is more leading and reaches the value of 32.1%. This indicates that our algorithm is accurate and stable in judging the motion trajectory of the vehicle during the continuous tracking of the vehicle. And while it's unfortunate that some of the methods don't disclose the runtime frame rate, at least our method still has a better score in terms of runtime speed. Even when compared to the latest algorithms, our algorithm remains advantageous. In summary, the improved algorithm in this paper has a high overall accuracy,

and it even reaches the highest level in the MT value that can represent the continuous tracking ability. It meets the real-time requirement while maintaining accuracy and tracking stability.

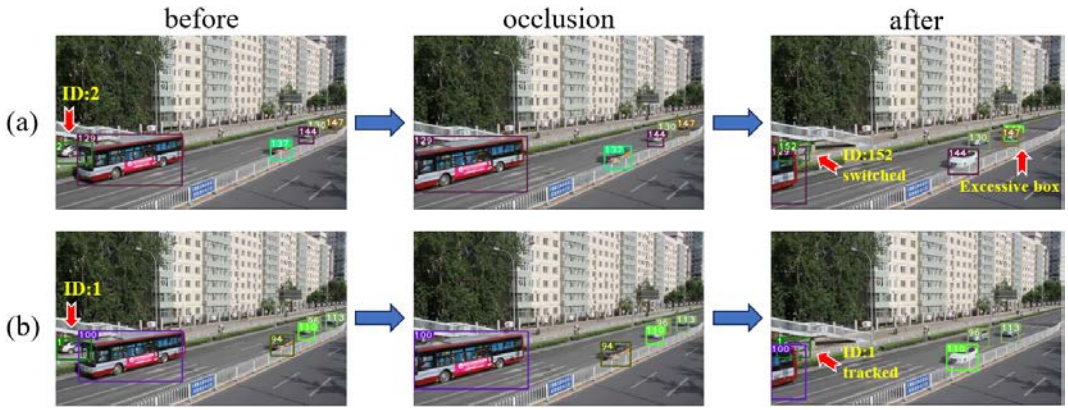


Fig. 11. Example of ID switch

Table 7. Comparison of tracking algorithms

Method	detector	MOTA	MT	FPS
SiamIOU [49]	EB	21.5	23.0	20.1
FAMNet [50]	CompACT	19.8	18.2	-
JDE [46]	Transformer	22.7	21.1	24.4
AttentionTrack [51]	-	24.0	17.8	26.3
DSLCF [52]	Mask RCNN	**30.3**	30.2	-
deepFAN [53]	EB	23.4	17.5	12.7
DeepSort [54]	YOLOv5-NAM	30.2	29.8	25.6
ours	YOLOv5	29.6	**32.1**	**31.4**

4.4 Discussion

Our experiments start from the detector and show that the detection accuracy gradually increases with the increase of improvement points, and finally, the mAP.50 value reaches 76%. This shows that our improvement has some relevance and has achieved appropriate results. The results of the real scene test show that some large vehicle missed detection problems, background misdetection problems, and anchor box shape problems, which exist in the baseline model, have been improved accordingly in our model. The reason for this is the focus of the C3_Res2A module on multi-scale targets and the improvement of the anchor box position by WIoU. The tracker, in turn, is based on the detector. A stable detection result provides correct information to the tracker. The improved ReID network also allows the tracker to associate this vehicle information more easily. The presented figure of results and data show that our model improves the ID switch problem to some extent and increases in the respective indexes.

However, due to the limitations of the model and the way the dataset was labeled, our algorithm still has some difficulty in recognizing smaller targets at long distances. When dealing with high-resolution images, our model may not handle the detailed information deep in the image well enough. In addition, even though the detection speed is not reduced, the large distance of objects between frames still affects the results when dealing with fast vehicles.

5. Conclusion

Multi-object tracking of vehicles has been one of the most popular research directions in the field of transportation nowadays and has been widely applied in many fields. In this paper, according to the realistic application scenarios, we fuse the multi-scale feature extraction module C3_Res2 with the lightweight attention SGE, and introduce the WIoU as a new loss function, which enhances the localization and recognition ability of vehicles by the YOLOv5s model. Secondly, by modifying the ReID network of DeepSort, makes it possible to judge the features more accurately when re-identifying the vehicle, without missing information, the association with the vehicle target of the previous frame is also more accurate. Finally, combined with the improved algorithm, several experiments were conducted on real scenarios and the UA-DETRAC dataset. The experiments have shown that the algorithm in this paper has improved the accuracy by 5.7% in vehicle detection. And combined with the tracking algorithm, it improves 6.5% and 4.7% in MOTA and MT values, and can significantly reduce the number of vehicle ID switching. Enhanced stability and reliability of vehicle tracking. On this basis, the running speed of the algorithm is not significantly changed but even slightly improved, which also satisfies the real-time requirements in real scenarios.

In the future, depending on the limitations of the algorithm as well as the task requirements, the speed can be further optimized and the size can be compressed to fit the hardware deployment. Improvements can also be made to the model's detail capture capability to adapt to distant target detection in traffic scenarios and to try to implement other practical functions.

Acknowledgement

This work was supported by the grants from the National Natural Science Foundation of China under No. 6230147. Jiangsu Province Industry-University-Research Co-operation Project under No. BY20231068, No. BY20231069. Jiangsu Province Higher Education Teaching Reform Research Project Funded Project under NO. 2023JSJG399. The Jiangsu Graduate Practical Innovation Project under NO. SJCX23_1871, NO. SJCX24_2153, NO. SJCX24_2153. Research on the Teaching Reform of High-quality Public Courses in Jiangsu Universities' funded project under NO. 2022JDKT110. Top-notch Academic Programs Project of Jiangsu Higher Education Institutions.

References

[1] F. Li, C.-H. Lee, C.-H. Chen et al., "Hybrid data-driven vigilance model in traffic control center using eye-tracking data and context data," *Advanced Engineering Informatics*, vol.42, 2019. Article (CrossRef Link)

[2] D. Song, R. Tharmarasa, G. Zhou, M. C. Florea, N. Duclos-Hindie and T. Kirubarajan, "Multi-Vehicle Tracking Using Microscopic Traffic Models," *IEEE Transactions on Intelligent Transportation Systems*, vol.20, no.1, pp.149-161, 2019. Article (CrossRef Link)

[3] Y. Zhang, B. Song, X. Du and M. Guizani, "Vehicle Tracking Using Surveillance With Multimodal Data Fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol.19, no.7, pp.2353-2361, 2018. Article (CrossRef Link)

[4] X. Cheng, D. Duan, L. Yang and N. Zheng, "Societal Intelligence for Safer and Smarter Transportation," *IEEE Internet of Things Journal*, vol.8, no.11, pp.9109-9121, 2021. Article (CrossRef Link)

[5] S. Sun, Y. Yin, X. Wang and D. Xu, "Robust Visual Detection and Tracking Strategies for Autonomous Aerial Refueling of UAVs," *IEEE Transactions on Instrumentation and Measurement*, vol.68, no.12, pp.4640-4652, 2019. Article (CrossRef Link)
[6] O. D. Jimoh, L. A. Ajao, O. O. Adeleke and S. S. Kolo, "A Vehicle Tracking System Using Greedy Forwarding Algorithms for Public Transportation in Urban Arterial," *IEEE Access*, vol.8, pp.191706-191725, 2020. Article (CrossRef Link)
[7] J. Azimjonov, A. Özmen, "A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways," *Advanced Engineering Informatics*, vol.50, 2021. Article (CrossRef Link)
[8] H. David and T. A. Athira, "Improving the Performance of Vehicle Detection and Verification by Log Gabor Filter Optimization," in *Proc. of 2014 Fourth International Conference on Advances in Computing and Communications*, pp.50-55, 2014. Article (CrossRef Link)
[9] G. Yan, M. Yu, Y. Yu et al., "Real-time vehicle detection using histograms of oriented gradients and AdaBoost classification," *Optik*, vol.127, no.19, pp.7941-7951, 2016. Article (CrossRef Link)
[10] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol.86, no.11, pp.2278-2324, 1998. Article (CrossRef Link)
[11] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.770-778, 2016. Article (CrossRef Link)
[12] A. G. Howard, M. Zhu, B. Chen et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv preprint arXiv:1704.04861*, 2017. Article (CrossRef Link)
[13] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *Proc. of 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.580-587, 2014. Article (CrossRef Link)
[14] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.779-788, 2016. Article (CrossRef Link)
[15] W. Liu, D. Anguelov, D. Erhan et al., "SSD: Single Shot MultiBox Detector," in *Proc. of 14th European Conference on Computer Vision – ECCV 2016, Part I*, vol.9905, pp.21-37, 2016. Article (CrossRef Link)
[16] Z. Zakria, J. Deng, R. Kumar, M. S. Khokhar, J. Cai and J. Kumar, "Multiscale and Direction Target Detecting in Remote Sensing Images via Modified YOLO-v4," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol.15, pp.1039-1048, 2022. Article (CrossRef Link)
[17] Y . Zhang, D. Liu et al., "Location First Non-Maximum Suppression for Uncovered Muck Truck Detection," *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, vol.E106.A, no.6, pp.924-931, 2023. Article (CrossRef Link)
[18] Z. Chen et al., "Fast vehicle detection algorithm in traffic scene based on improved SSD," *Measurement*, vol.201, 2022. Article (CrossRef Link)
[19] M. A. Butt, F. Riaz, "CARL-D: A vision benchmark suite and large scale dataset for vehicle detection and scene segmentation," *Signal Processing: Image Communication*, vol.104, 2022. Article (CrossRef Link)
[20] C.-Y. Wang, I.-H. Yeh, H.-Y. M. Liao, "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information," *arXiv preprint arXiv:2402.13616*, 2024. Article (CrossRef Link)
[21] Glenn J. et al., 2023. https://github.com/ultralytics/ultralytics
[22] G. Boesch, YOLOv7: A Powerful Object Detection Algorithm (2024 Guide). YOLOv7: The Fastest Object Detection Algorithm (2024) - viso.ai (accessed on 21 November 2023).
[23] C.-Y. Wang, A. Bochkovskiy and H.-Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," in *Proc. of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.7464-7475, 2023. Article (CrossRef Link)

[24] Glenn J., v7.0 - YOLOv5 SOTA Realtime Instance Segmentation, Nov. 2022. https://github.com/ultralytics/yolov5/releases
[25] L. Ma et al., "Visual localization with a monocular camera for unmanned aerial vehicle based on landmark detection and tracking using YOLOv5 and DeepSORT," *International Journal of Advanced Robotic Systems*, vol.20, no.3, 2023. Article (CrossRef Link)
[26] A. R. C. Caballo, and C. J. Aliac, "YOLO-based Tricycle Counting in Aid of Traffic Analysis," in *Proc. of the 2022 4th Asia Pacific Information Technology Conference*, pp.150-154, 2022. Article (CrossRef Link)
[27] D. N.-N. Tran, L. H. Pham, H.-H. Nguyen and J. W. Jeon, "City-Scale Multi-Camera Vehicle Tracking of Vehicles based on YOLOv7," in *Proc. of 2022 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, pp.1-4, 2022. Article (CrossRef Link)
[28] S. Dey, S. Winter, M. Tomko and N. Ganguly, "Traffic Count Estimation at Basis Links Without Path Flow and Historic Data," *IEEE Transactions on Intelligent Transportation Systems*, vol.24, no.10, pp.11410-11423, 2023. Article (CrossRef Link)
[29] H. Yuan and X. Song, "A Modified EKF for Vehicle State Estimation With Partial Missing Measurements," *IEEE Signal Processing Letters*, vol.29, pp.1594-1598, 2022. Article (CrossRef Link)
[30] J. Song, S.-H. Hyun, J.-H. Lee, J. Choi and S.-C. Kim, "Joint Vehicle Tracking and RSU Selection for V2I Communications With Extended Kalman Filter," *IEEE Transactions on Vehicular Technology*, vol.71, no.5, pp.5609-5614, 2022. Article (CrossRef Link)
[31] Q. Wang et al., "Transformer-Based Multiple-Object Tracking via Anchor-Based-Query and Template Matching," *Sensors*, vol.24, no.1, 2023. Article (CrossRef Link)
[32] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, "Simple online and realtime tracking," in *Proc. of 2016 IEEE International Conference on Image Processing*, pp.3464-3468, 2016. Article (CrossRef Link)
[33] N. Wojke, A. Bewley and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. of 2017 IEEE International Conference on Image Processing (ICIP)*, pp.3645-3649, 2017. Article (CrossRef Link)
[34] Z. Tong et al., "Wise-IoU: Bounding Box Regression Loss with Dynamic Focusing Mechanism," *arXiv preprint arXiv:2301.10051*, 2023. Article (CrossRef Link)
[35] J. Hu, L. Shen and G. Sun, "Squeeze-and-Excitation Networks," in *Proc. of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.7132-7141, 2018. Article (CrossRef Link)
[36] S. Woo et al., "CBAM: Convolutional Block Attention Module," in *Proc. of 15th European Conference on Computer Vision – ECCV 2018*, vol.11211, pp.3-19, 2018. Article (CrossRef Link)
[37] X. Li, X. Hu, and J. Yang, "Spatial Group-wise Enhance: Improving Semantic Feature Learning in Convolutional Networks," *arXiv preprint arXiv:1905.09646*, 2019. Article (CrossRef Link)
[38] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang and P. Torr, "Res2Net: A New Multi-Scale Backbone Architecture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.43, no.2, pp.652-662, 2021. Article (CrossRef Link)
[39] A. Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *Proc. of International Conference on Learning Representations (ICLR 2021)*, 2021. Article (CrossRef Link)
[40] L. Wen et al., "UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking," *Computer Vision and Image Understanding*, vol.193, 2020. Article (CrossRef Link)
[41] X. Liu et al., "A Deep Learning-Based Approach to Progressive Vehicle Re-identification for Urban Surveillance," in *Proc. of 14th European Conference on Computer Vision – ECCV 2016, Part II*, vol.9906, pp.869-884, 2016. Article (CrossRef Link)
[42] L. Yang et al., "SimAM: A Simple, Parameter-Free Attention Module for Convolutional Neural Networks," in *Proc. of 38th International Conference on Machine Learning, PMLR*, vol.139, pp.11863-11874, 2021. Article (CrossRef Link)

[43] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.39, no.6, pp.1137-1149, Jun. 2017. Article (CrossRef Link)
[44] A. Bochkovskiy, C.-Y. Wang, H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv preprint arXiv:2004.10934*, 2020. Article (CrossRef Link)
[45] L.-Y. Hao et al., "Multi-target vehicle detection based on corner pooling with attention mechanism," *Applied Intelligence*, vol.53, pp.29128-29139, 2023. Article (CrossRef Link)
[46] Z. Zhao, Z. Ji, Y. Yao, Z. He and C. Du, "Enhanced Detection Model and Joint Scoring Strategy for Multi-Vehicle Tracking," *IEEE Access*, vol.11, pp.30807-30818, 2023. Article (CrossRef Link)
[47] L. Wang et al., "DV3-IBi_YOLOv5s: A Lightweight Backbone Network and Multiscale Neck Network Vehicle Detection Algorithm," *Sensors*, vol.24, no.12, 2024. Article (CrossRef Link)
[48] L. Kang et al., "YOLO-FA: Type-1 fuzzy attention based YOLO detector for vehicle detection," *Expert Systems with Applications*, vol.237, part.B, 2024. Article (CrossRef Link)
[49] A. Li, L. Luo and S. Tang, "Real-Time Tracking of Vehicles with Siamese Network and Backward Prediction," in *Proc. of 2020 IEEE International Conference on Multimedia and Expo (ICME)*, pp.1-6, London, UK, 2020. Article (CrossRef Link)
[50] P. Chu and H. Ling, "FAMNet: Joint Learning of Feature, Affinity and Multi-dimensional Assignment for Online Multiple Object Tracking," in *Proc. of 2019 IEEE/CVF International Conference on Computer Vision*, pp.6172-6181, 2019. Article (CrossRef Link)
[51] C. Zhang, S. Zheng, H. Wu, Z. Gu, W. Sun and L. Yang, "AttentionTrack: Multiple Object Tracking in Traffic Scenarios Using Features Attention," *IEEE Transactions on Intelligent Transportation Systems*, vol.25, no.2, pp.1661-1674, 2024. Article (CrossRef Link)
[52] X. Hou, Y. Wang and L.-P. Chau, "Vehicle Tracking Using Deep SORT with Low Confidence Track Filtering," in *Proc. of 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp.1-6, 2019. Article (CrossRef Link)
[53] A. Prasannakumar, and D. Mishra, "Deep Efficient Data Association for Multi-Object Tracking: Augmented with SSIM-Based Ambiguity Elimination," *Journal of Imaging*, vol.10, no.7, 2024. Article (CrossRef Link)
[54] J. Wang, Y. Dong, S. Zhao, Z. Zhang, "A High-Precision Vehicle Detection and Tracking Method Based on the Attention Mechanism," *Sensors*, vol.23, no.2, 2023. Article (CrossRef Link)

Xiaole Ge received B.S. degrees from the Tongda College of Nanjing University of Posts & Telecommunications, China, in 2022. And now, he is pursuing the M.Eng. degree with the Yancheng Institute of Technology, Yancheng, China. His research topics include computer version technology and image processing.

Feng Zhou received the B.S. and M.S. degrees from Southeast University, Nanjing, China, in 2004 and 2012, respectively. He was awarded Associate Professor and Professor at the School of Information Engineering, Yancheng Institute of Technology in 2017 and 2023, respectively. His research interests are in machine vision, target detection and recognition, intelligent transport, remote sensing image processing.
Email: zfycit@ycit.edu.cn

Shuaiting Chen received B.S. degrees from China Jiliang University College of Modern Science and Technology in 2022. And now, he is pursuing the M.Eng. degree with the Yancheng Institute of Technology, Yancheng, China. His research topics include computer version technology and Intelligent control system.

Gan Gao received B.S. degrees from the Tongda College of Nanjing University of Posts & Telecommunications, China, in 2021. And now, he is pursuing the M.Eng. degree with the Yancheng Institute of Technology, Yancheng, China. His research topics include computer version technology and signal detection.

Rugang Wang received the B.S. degrees from the Wuhan University of Technology, Wuhan, China, in 1999, the M.S. degrees from Jinan University, Guangzhou, China, in 2007, and the Ph.D. degrees from Nanjing University, Nanjing, China, in 2012. And now, he is a Professor with the College of Information Engineering, Yancheng Institute of Technology, Yancheng, China. His research topics include the image processing technology, optical communication networks.
Email: wrg3506@ycit.edu.cn