IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 29, 2020, accepted January 11, 2021, date of publication January 14, 2021, date of current version January 25, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051659

A Video-Based Method With Strong-Robustness for Vehicle Detection and Classification Based on Static Appearance Features and Motion Features

YUE CHEN AND WUSHENG HU
School of Transportation, Southeast University, Nanjing 210096, China

Corresponding author: Wusheng Hu (13705151633@163.com)

This work was supported by the National Natural Science Foundation of China under Project 41574022 (National Natural Science Foundation of China, FUNDREF 10.13039/501100001809, GRANT #(s) No. 41574022).

ABSTRACT Vehicle detection and classification plays an important role in intelligent transportation system. Compared with traditional detectors, the detection and classification based on traffic surveillance video shows a huge advantage in its flexibility and continuity. However, to get wide applicability and strong robustness, most current methods focus on improving the accuracy of detectors by adjusting network parameters constantly, or increasing the size of training sets, which challenges the collection and labeling of data, the performance of computers, the scope of application and so on. Moreover, the unique continuity characteristic of the video, which can be used to describe the motion features of vehicle, is often ignored. Take these facts into account, this paper proposed a video-based vehicle detection and classification method, which is based on static appearance features and motion features both. Four detectors of different performance were trained with small training sets, and the designed algorithms for the remove, selection and reorganization of detected objects contribute to obtaining the optimal results of detection and classification. The experiment results show that the proposed method is able to detect and classify vehicles with more than 0.95 accuracy dealing with different road environments.

INDEX TERMS Video-based, vehicle detection, vehicle classification, static appearance features, motion features, deep learning, intelligent transportation system.

I. INTRODUCTION

With the increase of traffic pressure in recent years, intelligent transportation system (ITS) becomes more and more important in real-time traffic monitoring, solving traffic congestion problems and improving traffic safety. One of the most important parts in building a strong and reliable ITS is to collect large-scale traffic information data efficiently and accurately [1], especially for traffic volume, traffic density and traffic speed, which are used to describe and reflect the nature of traffic flow, whereas the detection and classification of vehicles is one of the most basic tasks to obtain traffic data. Therefore, how to detect and classify vehicles efficiently and accurately is getting more and more attention.

In the past, vehicles were detected mainly based on dedicated hardware [2]. The traditional detectors can be roughly divided into two categories according to different principles: based on wave frequency and based on magnetic frequency. Wave-frequency-based detectors detect the frequency change of reflected wave when the vehicle passes by to perceive the vehicle, whereas the available wave types include ultrasonic wave, sound wave, infrared wave and microwave, etc. The latter is to sense the change of coil inductance through the induction probe, such as induction coil detector, magnetic detector and so on.

However, traditional detectors are often inconvenient for installation and maintenance [3] and cost a lot, and they are also sensitive to ambient temperature, or airflow change. Except for these, one of the most important disadvantages is its low update rate of traffic information. In comparison, video-based method for vehicle detection and classification shows great advantages. Traffic surveillance video consists of a series of static traffic images in the form of continuous frames, changing by more than 24 frames per second. By analyzing and processing the continuous traffic images, the detection and classification of moving vehicles can be realized efficiently. Moreover, with the advantages of easy installation and maintenance, low cost, and

The associate editor coordinating the review of this manuscript and approving it for publication was Yongjie Li.

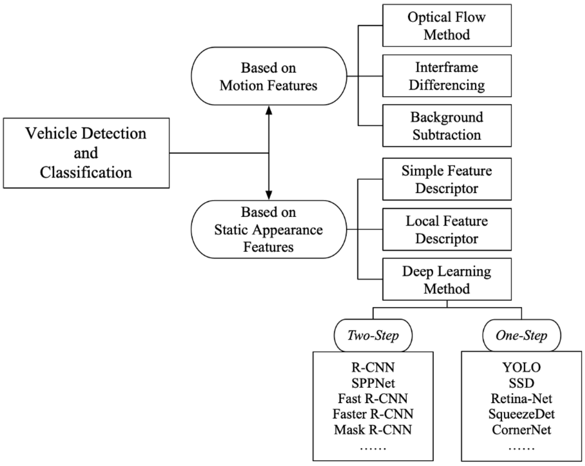


FIGURE 1. The summary of methods for vehicle detection and classification.

visualization, vehicle detection and classification based on video has attracted a lot of attentions in traffic information collection and becomes more and more important in the field of intelligent transportation.

In this paper, a vehicle detection and classification framework based on traffic video with wide applicability and strong robustness was proposed, and the static appearance features and motion features of vehicle were both applied in our system. To be specific, different detectors were trained on deep learning network with small training sets, which were based on the static appearance features and motion features respectively; at the same time, the algorithms for the remove, selection and reorganization of detected objects were designed to obtain the optimal results of detection and classification.

The rest of the paper is structured as follows: Section II introduces the related work of vehicle detection and classification. In Section III, we introduced a method for vehicle detection and classification. The training of different detectors was introduced in Section III-A, the remove of apparently false detection was in Section III-B, and the selection and reorganization of detected objects was in Section III-C. The experimental datasets and results were presented in Section IV. The discussion and conclusion were in Section V and Section VI.

II. RELATED WORK

At present, vehicle detection and classification technologies based on video can be generally divided into two categories: based on the inherent appearance features of static vehicles and based on motion features of moving vehicles. Many scholars have studied these two methods, and both methods have their own advantages and disadvantages. The summary of methods for vehicle detection and classification is shown in Fig. 1.

A. BASED ON MOTION FEATURES

Video is essentially a series of static and continuous images that change at a rate of more than 24 frames per second. Therefore, video has the advantage that a single static image does not, that is, continuous images are interrelated. In other words, a video contains the motion features of vehicle. Therefore, the principle of method based on motion feature is to use the correlation between images to segment the moving vehicle from the background in the form of binary image. In this case, the video meets the requirements of this method perfectly for its high image refresh rate. Moreover, the motion-based method does not require any prior knowledge but based on a set of images to identify the vehicle with a fast-running speed, which greatly improve its applicability and flexibility.

By introducing the effects of ego and relative motion [4], moving vehicles on the road can be extracted in the form of binary images with a segment of foreground and background. Optical flow method [5] has been used in vehicle detection for a long time, which is based on optical flow [6] calculations, and spatial features are utilized in this method. The optical flow algorithm is suitable for multi-target motion analysis, and the phenomenon of image block and overlap can be avoided [7], but its stability is poor, so it is not suitable for complex traffic environment. Therefore, some methods of image difference were proposed. Interframe differencing [8] is based on the difference of two or more successive image frames, whereas background subtraction is based on the difference of moving vehicles and stationary background [2]. As for the classification, it could be realize by giving the geometric features or binary features to different classifiers, such as support vector machine (SVM), artificial neural network (ANN), and AdaBoost [9].

Although the motion-based method is fast, it is seriously affected by constant changes in background, environment, video noise or other factors, the phenomenon of vehicle hole and unnecessary noise are serious. At the same time, its ability to classify vehicles is limited, for it is mostly based on the area of the segmented foreground.

B. BASED ON APPEARANCE FEATURES

Appearance-based methods are more intuitive and accurate, but require a lot of prior knowledge.

1) SIMPLE FEATURE DESCRIPTOR

In earlier methods, vehicle feature extraction is usually based on one or several features of vehicle, such as contour [10], texture [11], edge [12], color [13], or some parts of vehicle, such as windshield [14], lights [15], license plates [16] and so on. This method is very simple but not effective, because good feature extraction is often difficult to be obtained based on the simple description.

2) LOCAL FEATURE DESCRIPTOR

With the deepening of study, some methods based on local features [3] were proposed. This kind of method is to extract the vehicle by constructing some local feature descriptors [3]. For example, Histograms of Oriented Gradient (HOG) [17] is based on the evaluation of highly normalized local histogram of image gradient direction in a dense grid, whereas Harr-like feature descriptor is based on Haar basis functions [18] and was proposed by Viola and Jones [19] first. Compared with the previous simple feature descriptor, the robustness of this method was significantly improved.

3) DEEP LEARNING METHOD

In recent years, the method of deep learning (DL) has developed rapidly in the aspect of target detection, which shows a strong feature extraction ability and greatly improves the detection. Convolutional Neural Network (CNN) is one of the most successful applications and the AlexNet proposed by Krizhevsky *et al.* [20] had shown superior performance compared to previous approaches.

On this basis, some two-step detection methods were put forward step by step to improve the speed or accuracy, such as Spatial Pyramid Pooling Network (SPPNet), Region-based Convolutional Neural Network (R-CNN), Fast Region-based Convolutional Neural Network (Fast R-CNN), Faster Region-based Convolutional Neural Network (Faster R-CNN), Mask Region-based Convolutional Neural Network (Mask R-CNN) and so on. By introducing a SPP layer [21], SPPNet allows CNN model to generate a fixed length sequence to realize the calculation of feature map for the entire image only once, which speeds up the process greatly. Similarly, R-CNN [22] transforms the target region into a fixed image size and uses a selective search method. Whereas Fast R-CNN [23] allows us to train both the detector and the boundary box regression, which is an improvement of SPPNet and R-CNN. By introducing a network called Region Proposal Network (RPN), Faster R-CNN [24] could realize the simultaneous implementation of regional proposal generation and detection tasks. Whereas Mask R-CNN [25] is an extension of Faster R-CNN to solve the instance segmentation problem, but it also adds some computational overhead to the network.

To further speed up the algorithm, one-step detection method was proposed, such as You Only Look Once (YOLO) [26], Single Shot Multi-Box Detector (SSD) [27], Retina-Net [28], SqueezeDet [29], CornerNet [30], etc. Instead of using regions to locate targets in two-step methods, the one-step applies the entire image to a CNN. This method divides the whole image into regions and predicts the boundary box and probability of each region. However, we have to admit that the increase of speed comes at the expense of the decrease of accuracy to some extent. Accuracy and speed often cannot be both, just as the accuracy of a simple feature descriptor is far lower than a DL method.

C. SHORTCOMINGS OF PREVIOUS METHODS

1) BASED ON MOTION FEATURES

As mentioned in Section II-A, motion-based method makes good use of the motion features of vehicles and has a fast-computing speed, but the results are of lower accuracy. Moreover, the ability to classify vehicles is often limited.

2) BASED ON APPEARANCE FEATURES

- *High Requirements for Prior Knowledge and Computer Performance*

For the appearance-based method, it is more intuitive, but a lot of prior knowledge and repeated training are needed to obtain a high-precision detector. As is known to all, the high performance of feature detector is often achieved through a large amount of prior knowledge and complex training models, which brings certain challenges to the size of training set, the high performance of computer and the training time. For example, in [31], 6,467 images were selected for Faster R-CNN training on a computer workstation with Intel(R) i7-8700 @3.20 GHz CPU, 16 GB of RAM and a GTX 1060Ti GPU); in [32], over 10,000 vehicle images were put into Mask R-CNN to train the feature detector on a desktop computer with Intel i9-7980XE (18 cores and 36 threads) @ 4.2Hz, 64 GB DDr4 (3200MHz) memory and two Nvidia 2080ti GPUs; whereas in [33], 83,791 images were used for training Yolo-v3 model on four NVIDIA GeForce GTX TITAN XP GPU with 12GB memory. Moreover, labeling a large number of images is also a bulky work.

- *Difficult Selections of Appropriate Training Set, Training Network and Parameters*

For different road environments, the selection of training set, training network and parameters should be adjusted separately according to different road environments.

For the training set, however, the more training images are not necessarily the better, because too many training images are prone to over-fitting, which will directly lead to the greatly reduced application scope of the detector. The size of training set should be adjusted according to the network hierarchy we built. Take the network depth and number of parameters as examples to illustrate. As shown in Table 1, the depth and number of parameters in different networks vary greatly, so the sizes of training set training with different networks are naturally different. At the same time, the quality of image used for training is very important, which will also affect the reliability of a detector.

TABLE 1. The depth and number of parameters in different networks.

Network	Depth	Parameters (Millions)
AlexNet [20]	8	61
VGG16 [36]	16	138
VGG19 [36]	19	144
GoogLeNet [37]	22	7
ResNet18 [38]	18	11.7
ResNet50 [38]	50	25.6
ResNet101 [38]	101	41.6

Likewise, the network selection and parameter adjustment for training are also very important. The inputs [34] that need to be adjusted may include the initial learning rate, the way for updating the learning rate, the contribution of previous step, the gradient threshold, etc. For the adjustment of parameters, for example, the choose of gradient descent optimization algorithm [35] could be the stochastic

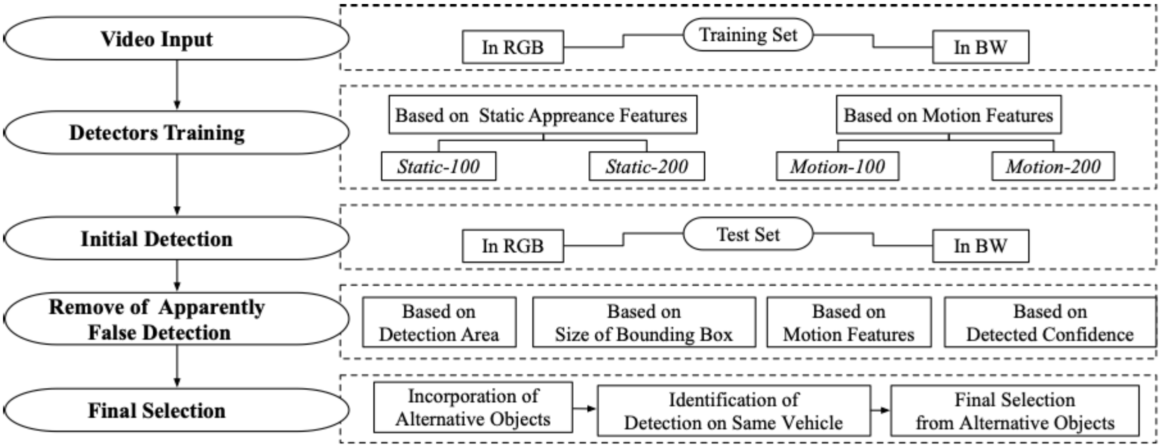

Video Input
Training Set
In RGB
In BW
Detectors Training
Based on Static Appreance Features
Based on Motion Features
Static-100
Static-200
Motion-100
Motion-200
Initial Detection
Test Set
In RGB
In BW
Remove of Apparently False Detection
Based on Detection Area
Based on Size of Bounding Box
Based on Motion Features
Based on Detected Confidence
Final Selection
Incorporation of Alternative Objects
Identification of Detection on Same Vehicle
Final Selection from Alternative Objects

FIGURE 2. Block diagram of the proposed framework.

gradient descent with momentum (SGDM), or the root mean square prop (RMSprop), whereas the corresponding parameters may include batch size, momentum, weight decay, and so on.

Unfortunately, however, there is no standard to measure how well a network is set up, which further increases the difficulty of detector training. Therefore, in order to improve the reliability of detectors, what we could do may be keeping trying and researching to get better rather than the best results. Moreover, the detection is only based on static images, which is a waste from a video perspective.

Based on the shortcomings introduced above, a vehicle detection and classification method that requires less prior knowledge for deep learning was proposed. At the same time, different from previous studies, we did not focus on the training of high-precision detectors. Instead, we combined the detected results of different detectors trained with appearance and motion features respectively, and designed algorithms for the remove, selection and reorganization of detected objects to realize high robustness and a wide range of applications for vehicle detection and classification.

III. METHODOLOGY

The proposed method is based on traffic surveillance video, and the block diagram of proposed framework is shown in Fig. 2.

A. DETECTORS TRAINING BASED ON STATIC APPEARANCE FEATURES AND MOTION FEATURES

1) SELECTION OF TRAINING NETWORK

Among the three appearance-based methods mentioned in Section II-B, DL method is the one with the highest accuracy and the best detection effect, therefore, it was chosen to obtain the feature detectors in this paper. Compared with the lower accuracy of one-step method, a two-stage detector may be a better choice. Whereas as shown in Fig. 1, Mask R-CNN may be a more advanced two-stage method, but it also adds some computational overhead to the network. Therefore, considering the detection accuracy and training speed comprehensively, we chose Faster R-CNN to train the feature detectors.

As for the selection of backbone for training Faster R-CNN, many mature pre-trained CNN network could be chosen from, such as AlexNet, VGG, GoogLeNet, ResNet and so on. The first to catch our attention was AlexNet designed by Krizhevsky *et al.* [20], which won the first prize due to its excellent performance in ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 [39]. Compared with other up-and-comers, AlexNet has a lower accuracy, but its speed is relatively faster. To compare the performance of different networks in terms of accuracy and running time, we carried out the experiments of training detector with different backbones. Based on the proposed method introduced in the next sections, the accuracy and running time of different backbones were calculated respectively. Using the AlexNet as a benchmark, we measured the improvement in accuracy and the increase in running time. As shown in Table 2, compared to AlexNet, the improved accuracy of other backbones is limited, but the time consumption has increased significantly. Therefore, considering the two factors of accuracy and running time comprehensively, AlexNet was selected as the backbone for training Faster R-CNN in this study.

TABLE 2. The performance of different networks.

	VGG16	VGG19	GoogLeNet	ResNet18	ResNet50	ResNet101
Accuracy Improved	0.157%	0.167%	0.111%	0.139%	0.185%	0.204%
Time Increased	437.7%	657.8%	164.9%	114.9%	206.5%	332.7%

2) SELECTION OF TRAINING SET WITH SMALL SIZE

Different from previous studies, we no longer focus on how to improve the reliability of one detector continuously, but train different detectors based on different small training sets, and realize high-precision vehicle detection by utilizing the complementarity of performance of different detectors.

For the selection of training set, compared with highway, the environment of urban road is more complicated, and the difficulty of detection and classification is also increased. Therefore, a city traffic surveillance video was chosen as the source of our training set. The selected video, named Video I, was recorded by ourselves in Nanjing, Jiangsu, China, and can be obtained from [40]. It is a five-minute video with a frame rate of 30 fps, so this video contains 9,000 frames. In this five-minute video, there are 90 vehicles (1,080 v/h) and 16 are large, accounting for 17.78%.

Two static detectors will be trained based on static appearance features with RGB image, and two motion detectors will be trained based on motion features with binary image in this paper. The training sets of the static detectors and the motion detectors are the same, except that the former ones are in RGB, and the latter ones are in binary.

Two small training sets were constructed for training, which contained only 100 and 200 images respectively, named Training-100 and Training-200. In order to make the image contain more features, a selection rule was designed. The 9,000 frames were divided into 100 groups averagely in chronological order first, so that each group contains 90 images. The second step is for image selection. To illustrate the selection better, take the selection of Training-200 as an example. Two images will be selected randomly from each of these 100 groups. In each selected frame, at least one vehicle should be contained. If not, a new image will be randomly selected from the same group until the new selected frame contains at least one vehicle. Finally, the 200 selected frames will be randomly scrambled to form Training-200. Training-100 was selected using the same rule, except that in second step, only one image will be selected from each group.

Therefore, based on Training-100 in RGB and Training-200 in RGB, two static detectors could be trained, and we named them as *Static*-100 and *Static*-200. Likewise, the two motion detectors could be named as *Motion*-100 and *Motion*-200.

3) CLASSIFICATION OF VEHICLE

In the previous research on vehicle classification, the focus was on how to divide vehicles into as many classes as possible in order to obtain more detailed results. For example, Sun and Ritchie [41] divided vehicles into seven different types. Although under this kind of classification, the results were detailed. However, with the development of automobile industry, there are more and more vehicle styles and shapes, and sometimes it is difficult to distinguish different the class of a vehicle accurately and meticulously by our human, not to mention machine learning. Moreover, to collect so many types of vehicles in actual traffic environments is not an easy task, and higher requirements will also be put forward for the size of training set. At the same time, according to [42], considering that different types of vehicles have different effects on traffic, vehicles are categorized into two types (passenger car and heavy vehicle) to analyze the traffic. Therefore, taking the above factors into consideration, we no longer classified vehicles in detail, but divided them into two big categories: small vehicles and large vehicles in this study.

However, the classification did not stop there. Due to the continuity of image sequence, for the vehicle detection and classification, there is a big difference between a video and a single and isolated image, that is, there are many incomplete vehicles in a video. Just like Fig. 3 shown, each vehicle in the video will go through a process from complete (Fig. 3a,c) to incomplete (Fig. 3b,d) before disappearing from the image, and the features of complete and incomplete vehicles are very different. Therefore, complete vehicles and incomplete vehicles should be regarded as different classes. In this case, vehicles were divided into four classes, namely complete small vehicle (CS), incomplete small vehicle (IS), complete large vehicle (CL) and incomplete large vehicle (IL), shown in Fig. 3.

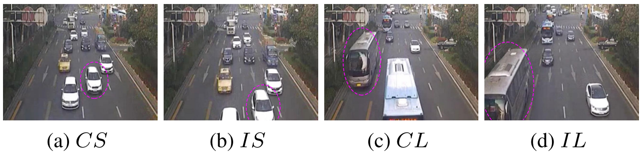

(a) CS (b) IS (c) CL (d) IL

FIGURE 3. Interpretation of vehicle classification. (a) Complete small vehicle (*CS*). (b) Incomplete small vehicle (*IS*). (c) Complete large vehicle (*CL*). (d) Incomplete large vehicle (*IL*).

4) TRAINING SET LABELING AND GROUND TRUTH DATA

To train a detector, images labeled with ground truth label data are necessary. The Image Labeler app—a MATLAB application, which provides an easy way to manually mark rectangle bounding box—was chosen to obtain the ground truth label data of training sets, whereas the obtained data generally contains information of data source, label definitions, and ground truth data according to the labeling. The output of labeled ground truth data in each Frame k includes the descriptions of the position, size, and category of each bounding box for each object, which could be expressed as:

$$\begin{aligned} Bbox_k &= \begin{bmatrix} X^{upper\text{-}left} & Y^{upper\text{-}left} & Width & Height & Class \end{bmatrix} \\ &= \begin{bmatrix} x_1^{upper\text{-}left} & y_1^{upper\text{-}left} & width_1 & height_1 & class_1 \\ x_2^{upper\text{-}left} & y_2^{upper\text{-}left} & width_2 & height_2 & class_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_M^{upper\text{-}left} & y_M^{upper\text{-}left} & width_M & height_M & class_M \end{bmatrix} \end{aligned} \tag{1}$$

where $X^{upper\text{-}left}$ and $Y^{upper\text{-}left}$ represent the horizontal and vertical coordinates of the upper left corner, *Width* and *Height* represent the width and height of the bounding box, and M represents the number of objects in Frame k.

It is worth noting that, the vehicles far away (upper part of the image) are too small, so it is not conducive to marking, training and subsequent identification. Therefore, vehicles that were too small in the distance were not marked in this study. Take the images in Fig. 3 as examples for the illustration of labeling vehicles with Image Labeler, the labeling results are shown in Fig. 4.

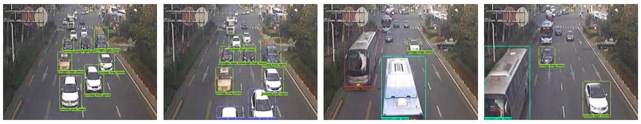

(a) CS (b) IS (c) CL (d) IL

FIGURE 4. Labeling results of the images shown in Fig. 3.

5) DETECTED DATA OF TEST SET

When detecting an image with trained detector, not only the position ($X^{upper\text{-}left}$, $Y^{upper\text{-}left}$), size ($Width$ and $Height$), category ($Class$) of each bounding box could be obtained, but also the confidence level ($Score$). Similarly, the four outputs could be unified into a matrix. In order to distinguish it from the $Bbox$ in training set, we named the matrix box, which could be expressed as:

$$
\begin{aligned}
box_k^{detector} &= \begin{bmatrix} X^{upper\text{-}left} & Y^{upper\text{-}left} & Width & Height & Class & Score \end{bmatrix} \\
&= \begin{bmatrix} x_1^{upper\text{-}left} & y_1^{upper\text{-}left} & width_1 & height_1 & class_1 & score_1 \\ x_2^{upper\text{-}left} & y_2^{upper\text{-}left} & width_2 & height_2 & class_2 & score_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^{upper\text{-}left} & y_N^{upper\text{-}left} & width_N & height_N & class_N & score_N \end{bmatrix}
\end{aligned} \tag{2}
$$

where $box_k^{detector}$ means the detected result of Frame k using Detector $detector$; whereas N is the total number of detected objects.

6) DETECTORS TRAINING BASED ON STATIC APPEARANCE FEATURES

The training of the two static appearance feature detectors ($Static$-100 and $Static$-200) were operated on Training-100 in RGB and Training-200 in RGB respectively. The SGDM algorithm [43] was utilized to update the weights with parameters of 0.9 momentum. The global training process was conducted for an epoch batch size of 20, and a maximum of 100 of iterations, whereas the initial learning rate was set to 10^{-4}.

To better illustrate our method, we selected a frame ($k = 979$) to show the results of each process in the form of an image, here we called its RGB form as I_k^{RGB}. In the selected frame, there are two complete small vehicles (CS), one incomplete small vehicle (IS), one complete large vehicle (CL) and one incomplete large vehicle (IL). $Static$-100 and $Static$-200 were used for vehicle detection and classification. Using the expression form of (2), we express the two initial results as $box\text{-}1_k^{Static\text{-}100}$ and $box\text{-}1_k^{Static\text{-}200}$. The serial number i, $Class$ and $Score$ of each detected bounding box were marked in I_k^{RGB} for easy observation, shown in Fig. 5a,b.

For $box\text{-}1_k^{Static\text{-}100}$, seven objects were detected, whereas two of them were a pair of detection on a same complete large bus ($box\text{-}1_k^{Static\text{-}100}{}_2$ and $box\text{-}1_k^{Static\text{-}100}{}_6$), and one was noise ($box\text{-}1_k^{Static\text{-}100}{}_7$). For $box\text{-}1_k^{Static\text{-}200}$, eight objects were detected, and there were three detection pairs on the same vehicle. More details are shown in Table 3. According to the analysis, it could be found that the detection results of $Static$-100 and $Static$-200 were different, which illustrated that different detectors have different recognition performances for a same image.

7) DETECTORS TRAINING BASED ON MOTION FEATURES

Different from the static appearance features reflected in RGB images intuitively, motion features cannot be directly obtained. Therefore, the training of motion feature detector is relatively complex, because the motion features should be extracted first.

Different from RGB images, a pixel of a binary image has only two values, namely 1 (white) and 0 (black). However, with these two values, a binary image could describe the position and shape of the vehicle, because it is the segmentation of the vehicle and the background. In this case, such binary segmentation image is a tool to extract the motion features.

The method based on the difference between stationary background and moving vehicle was used to obtain the binary images, and the background model was established with the statistical median method [44]. Since a binary image has only two values, compared with a RGB image, it is more intuitive to reflect the shape and position information of the vehicle.

The training of motion feature detectors was conducted on binary images. The selection and parameter setting of the training network are consistent with the training of static appearance feature detectors, whereas the classification of vehicle (CS, IS, CL and IL) and the two training sets are also the same. The difference is that the training sets need to be converted to binary images first.

Let us make a comprehensive analysis of the performance of the four detectors. As shown in Table 3, the detection results of each detector were different, which illustrated that different detectors have different recognition performance for a same image. At the same time, the error of missing detection seems more common for the motion detectors, whereas the errors of repeated detection on a same vehicle and noise seems more common for the static detectors, which further explained the different performance of detectors and

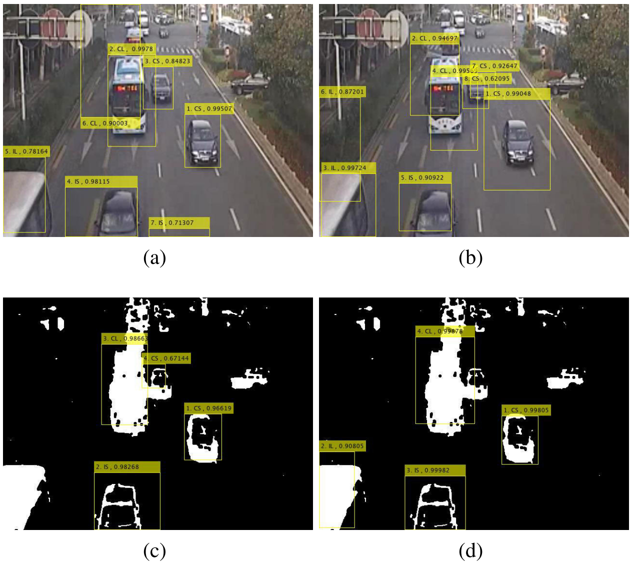
TABLE 3. Analysis of the performance of four detectors.

Detector	Total-Detected	Missed	Repeated-Pairs	Noise
$Static$-100	7	0	1 (**2&6**)	1 (**7**)
$Static$-200	8	0	3 (**2&4**; **3&6**; **7&8**)	0
$Motion$-100	4	1	0	0
$Motion$-200	4	1	0	0

their good complementarity. In the latter sections, how to utilize the complementarity of different detectors to improve the robustness and generalization ability of vehicle detection algorithm will be introduced.

B. REMOVE OF APPARENTLY FALSE DETECTION

As shown in Fig. 5, some detected objects were apparently false, such as objects beyond the driveways, objects whose bounding box size does not match the vehicle size, objects with low confidence, and so on. In this case, these kinds of object could be removed based on some rules.



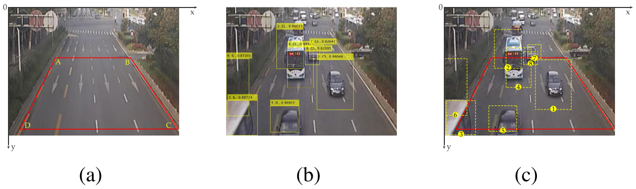
(a) (b) (c) (d)

FIGURE 5. Initial detection of the selected frame based on the four detectors. (a) $box\text{-}1_k^{Static\text{-}100}$. (b) $box\text{-}1_k^{Static\text{-}200}$. (c) $box\text{-}1_k^{Motion\text{-}100}$. (d) $box\text{-}1_k^{Motion\text{-}200}$.

1) REMOVE BASED ON DETECTION AREA

The detection area should be defined first. Based on the common sense that vehicles only driving on the road, the delineation of detection area was set roughly the same as the driveway. Moreover, as mentioned in Section III-A, vehicles in the distance are too small for marking and identification, so the detection area is reduced a little relative to the driveway. The detection area is shown in Fig. 6a. At the same time, it is important to note that, the origin of coordinates was set in the upper left corner in this study.

A rule for determining whether a vehicle is within the detection area was set, that is, as long as a part of the vehicle enters the detection area, the vehicle is considered to be within the detection area, not requiring all parts. As shown in (2), the position of Object i is described by the upper left corner $(x_i^{upper\text{-}left}, y_i^{upper\text{-}left})$. However, as stated in the rule above, the upper left corner of a bounding box is not applicable to describing the position of a box. Based on this, the midpoint of the lower boundary $(x_i^{lower\text{-}mid}, y_i^{lower\text{-}mid})$ was used to describe the position for determining whether a vehicle is in the detection area, which could be expressed as:



(a) (b) (c)

FIGURE 6. The interpretations of detection area and midpoint, and a remove example. (a) The detection area. (b) The midpoints of $box\text{-}1_k^{Static\text{-}200}$. (c) The remove of Object 6.

$$\begin{aligned} x_i^{lower\text{-}mid} &= x_i^{upper\text{-}left} + \frac{1}{2} \times width_i \\ y_i^{lower\text{-}mid} &= y_i^{upper\text{-}left} + height_i \end{aligned} \tag{3}$$

Take the initial detection of $box\text{-}1_k^{Static\text{-}200}$ as an example. The midpoints of each objects are marked in Fig. 6b.

As shown in Fig. 6a, the horizontal Boundary AB could be used as the upper boundary. Moreover, just like the $box\text{-}1_k^{Motion\text{-}100}{}_1$ shown in Fig. 5c, the bounding box could not completely cover the detected object in some cases, so it is necessary to give a certain floating interval when setting the upper boundary. In this case, whether $box_k^{detector}{}_i$ should be removed under the constraint of upper boundary, the following condition should be met:

$$remove_i^{upper} = \begin{cases} 1, & if \quad y_i^{lower\text{-}mid} \leq b_{AB} + T^{upper} \\ 0, & otherwise \end{cases} \tag{4}$$

where b_{AB} is the intercept of Boundary AB, whereas T^{upper} is the floating interval for upper boundary. As for the lower boundary, there is not much significance in setting, because Boundary CD is almost attached to the lower boundary of the image, that is, the disappearance of a bounding box in the image represents the departure of the vehicle.

Similarly, for the left and right boundary, Boundary AD and Boundary BC were taken as the boundaries, and the remove conditions based on left and right boundary could be expressed as:

$$\begin{aligned} &remove_i^{left} \\ &= \begin{cases} 1, & if \ y_i^{lower\text{-}mid} \leq (k_{AD} \times x_i^{lower\text{-}mid} + b_{AD}) + T^{left} \\ 0, & otherwise \end{cases} \end{aligned} \tag{5}$$

$$\begin{aligned} &remove_i^{right} \\ &= \begin{cases} 1, & if \ y_i^{lower\text{-}mid} \leq (k_{BC} \times x_i^{lower\text{-}mid} + b_{BC}) + T^{right} \\ 0, & otherwise \end{cases} \end{aligned} \tag{6}$$

where k_{AD} and k_{BC} are the slopes of Boundary AD and Boundary BC, whereas b_{AD} and b_{BC} are the intercepts. T^{left} and T^{right} are the floating intervals for left and right boundary respectively. As shown in Fig. 6b, $box\text{-}1_k^{Static\text{-}200}{}_6$ met the remove condition of left boundary, so it should be removed.

2) REMOVE BASED ON SIZE OF BOUNDING BOX

Generally speaking, the size of a vehicle is limited within a certain range. Therefore, the size of its bounding box is also limited in both width and height, which can be used as a prior knowledge to remove false detected objects.

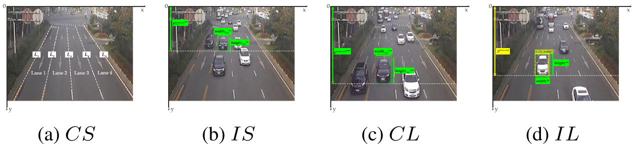
Before introducing the limit range of bounding box size, a method to determine which lane a vehicle belongs to should be given. As shown in Fig. 7a, a four-lane road is divided by five lane lines, that is L_1, L_2, L_3, L_4 and L_5. In this case, it is only necessary to know the slope and intercept of each lane line to determine which lane the vehicle belongs to. Similarly, the midpoint of the lower boundary of a box was used to determine the position of a detected object. Therefore, for Object i, the lane number ($lane$) which it belongs to could be expressed as:

$$lane = \begin{cases} 1, & if\ \dfrac{y_i^{lower\text{-}mid} - b_{L_1}}{k_{L_1}} \le x_i^{lower\text{-}mid} < \dfrac{y_i^{lower\text{-}mid} - b_{L_2}}{k_{L_2}} \\ 2, & if\ \dfrac{y_i^{lower\text{-}mid} - b_{L_2}}{k_{L_2}} \le x_i^{lower\text{-}mid} < \dfrac{y_i^{lower\text{-}mid} - b_{L_3}}{k_{L_3}} \\ 3, & if\ \dfrac{y_i^{lower\text{-}mid} - b_{L_3}}{k_{L_3}} \le x_i^{lower\text{-}mid} < \dfrac{y_i^{lower\text{-}mid} - b_{L_4}}{k_{L_4}} \\ 4, & if\ \dfrac{y_i^{lower\text{-}mid} - b_{L_4}}{k_{L_4}} \le x_i^{lower\text{-}mid} < \dfrac{y_i^{lower\text{-}mid} - b_{L_5}}{k_{L_5}} \end{cases} \tag{7}$$

where k_{L_1}, k_{L_2}, k_{L_3}, k_{L_4}, k_{L_5} and b_{L_1}, b_{L_2}, b_{L_3}, b_{L_4}, b_{L_5} represent the slope and intercept of each lane line.

Since the video is not shot vertically towards the road, the level of distortion varies for each position in the video. However, for the sake of research, in the horizontal direction, we only divide the difference into four segments based on these four lanes. As for the vertical direction, it is a little more complicated. According to the principle of linear propagation of light, in order to obtain the magnitude of a bounding box at any position of the lane, we only need to know its width and height at any two points. In this case, the slope and intercept could be calculated first before determining the standard height and width of a bounding box on the lane.

As mentioned before, vehicles were divided into four classes, that is complete small vehicle (CS), incomplete small vehicle (IS), complete large vehicle (CL) and incomplete large vehicle (IL). Therefore, different restrictions should be given to these four classes. For the complete vehicles (CS and CL), it could be expressed in (8), as shown at the bottom of the next page, where $size = width, height$; $class = CS, CL$; $lane = 1, 2, 3, 4$. $k_{size,class,lane}$ and $b_{size,class,lane}$ represent the slope and intercept of $width$ or $height$ for $class$ on $lane$ respectively. $size_{class,lane}$ and $y_{class,lane}^{lower\text{-}mid}$ should be determined respectively according to different $class$ and $lane$. As for the incomplete vehicles (IS and IL), only the width should be restricted, because the height of the incomplete is independent of the height of the vehicle itself. In this case, $k_{width,IS}$ and $b_{width,IS}$ equal to $k_{width,CS}$ and $b_{width,CS}$ for small vehicle, whereas $k_{width,IL}$ and $b_{width,IL}$ equal to $k_{width,CL}$ and $b_{width,CL}$ for large. As for $k_{height,IS}$, $b_{height,IS}$, $k_{height,IL}$ and $b_{height,IL}$, there is no need to calculate as explained above. In this case, the slope and intercept of incomplete vehicles could be expressed as:



(a) CS (b) IS (c) CL (d) IL

FIGURE 7. Interpretations of the lane, the illustration of calculating process of k and b, and an example. (a) The four lanes. (b) The vehicles in the distance. (c) The vehicles in the near. (d) The example.

$$\begin{aligned} k_{width,IS} &= k_{width,CS}, \quad b_{width,IS} = b_{width,CS} \\ k_{width,IL} &= k_{width,CL}, \quad b_{width,IL} = b_{width,CL} \\ k_{height,IS} &= 0, \quad b_{height,IS} = 0 \\ k_{height,IL} &= 0, \quad b_{height,IL} = 0 \end{aligned} \tag{9}$$

Here we use the example of calculating $k_{size,CS,2}$, $b_{size,CS,2}$, $k_{size,IS,2}$ and $b_{size,IS,2}$ to illustrate the calculation process. A complete small vehicle driving on Lane 2 in the distance (Fig. 7b) and in the near (Fig. 7c) was chosen randomly. What we need to obtain is $width_{CS,2}^{near}$, $height_{CS,2}^{near}$, ${y_{CS,2}^{lower\text{-}mid}}^{near}$ (from Fig. 7b), and $width_{CS,2}^{far}$, $height_{CS,2}^{far}$, ${y_{CS,2}^{lower\text{-}mid}}^{far}$ (from Fig. 7c). In this case, $k_{width,CS,2}$, $b_{width,CS,2}$, $k_{height,CS,2}$ and $b_{height,CS,2}$ could be calculated following (8). As for IS, according to (9), $k_{width,IS,2} = k_{width,CS,2}$, $b_{width,IS,2} = b_{width,CS,2}$, $k_{height,IS,2} = 0$, $b_{height,IS,2} = 0$.

Therefore, for Object i, the standard height and width of its bounding box could be determined with $k_{size,class,lane}$ and $b_{size,class,lane}$, which could be expressed as:

$$size_i^{std} = k_{size,class,lane} \times y_i^{lower\text{-}mid} + b_{size,class,lane} \tag{10}$$

where $size = width, height$; $class = CS, IS, CL, IL$; $lane = 1, 2, 3, 4$, whereas $lane$ is determined by (7). In this case, the size of bounding box i could be restricted by $width_i^{std}$ and $height_i^{std}$. At the same time, a certain amount of adjustment space should be given within the standard scope. Therefore, whether Object i should be removed, the following conditions should be met:

$$remove_i^{size} = \begin{cases} 1, & if\ \{size_i \le size_i^{std} \times T_{low}^{size}\} \\ & \quad \vee \{size_i \ge size_i^{std} \times T_{high}^{size}\} \\ 0, & otherwise \end{cases} \tag{11}$$

where $size = width, height$; $0 \le T_{low}^{size} \le 1 \le T_{high}^{size}$. T_{low}^{size} and T_{high}^{size} represents the minification and magnification of $width^{std}$ and $height^{std}$ respectively.

A complete small vehicle (CS) driving on Lane 2 was selected randomly as an example for illustration. The detected results were marked in yellow, whereas the $width_i^{std}$ and $height_i^{std}$ were marked in green. As shown in Fig. 7d, its $width_i$ and $height_i$ did not meet the remove conditions of size, so it should not be removed.

3) REMOVE BASED ON MOTION FEATURES

As we mentioned in Section III-A, the pixel value of 1 or 0 in a binary image can be used to judge the state of a point, static or moving. Therefore, the correctness of a detected object can be distinguished based on the pixel values of binary image. In this case, for Object i, the ratio of $pixel = 1$ in its bounding box should meet certain conditions, which could be expressed as:

$$motion_i = \frac{\sum pixel(x, y)}{width_i \times height_i} \tag{12}$$

where

$$\begin{aligned} x &\in [x_i^{upper\text{-}left}, x_i^{upper\text{-}left} + width_i] \\ y &\in [y_i^{upper\text{-}left}, y_i^{upper\text{-}left} + height_i] \end{aligned} \tag{13}$$

where $pixel(x, y)$ represents the pixel value at position (x, y) within its bounding box, whereas $motion_i$ represents the ratio of $pixel = 1$. Therefore, under the constraint of motion features, whether Object i should be removed, the following condition should be met:

$$remove_i^{motion} = \begin{cases} 1, & if\ motion_i \leq T^{motion} \\ 0, & otherwise \end{cases} \tag{14}$$

where T^{motion} is the threshold for the ratio of $pixel = 1$. Take the initial detection of $box\text{-}1_k^{Static\text{-}100}$ as an example. As shown in Fig. 5, it is obvious that $remove_7^{motion}$ equals to 1. In this case, ${box\text{-}1_k^{Static\text{-}100}}_7$ was noise and should be removed.

4) REMOVE BASED ON DETECTED CONFIDENCE

A detected object with a low confidence level should be rejected. Therefore, under the constraint of detected confidence, whether Object i should be removed, the following condition should be met:

$$remove_i^{score} = \begin{cases} 1, & if\ score_i \leq T^{score} \\ 0, & otherwise \end{cases} \tag{15}$$

where T^{score} is the threshold for the detected confidence. Take the initial detection (Fig. 5) of the selected test frame as examples. The detected confidences of the initial detection are listed in Table 4. With $T^{score} = 0.8$ as threshold, the objects shown in **bold** should be removed.

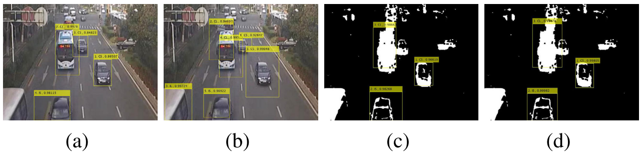
Through the four kinds of constraints above, whether Object i should be removed preliminarily, the following conditions should be met:

$$remove_i = \begin{cases} 1, & if \sum remove_i^{basis} \geq 1 \\ 0, & otherwise \end{cases} \tag{16}$$

TABLE 4. Detected confidences of the initial detection.

Detector	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$
$Static$-100	0.995	0.998	0.848	0.981	**0.787**	0.900	**0.713**	-
$Static$-200	0.991	0.947	0.997	0.996	0.909	0.872	0.927	**0.621**
$Motion$-100	0.966	0.983	0.987	**0.671**	-	-	-	-
$Motion$-200	0.998	0.908	0.999	0.998	-	-	-	-

where $basis = upper, left, right, width, height, motion, score$. The remove results ($box\text{-}2_k^{detector}$) of the selected test frame are shown in Fig. 8.



(a) (b) (c) (d)

FIGURE 8. Remove results of the selected frame. (a) $box\text{-}2_k^{Static\text{-}100}$. (b) $box\text{-}2_k^{Static\text{-}200}$. (c) $box\text{-}2_k^{Motion\text{-}100}$. (d) $box\text{-}2_k^{Motion\text{-}200}$.

C. FINAL SELECTION

1) INCORPORATION OF ALTERNATIVE OBJECTS

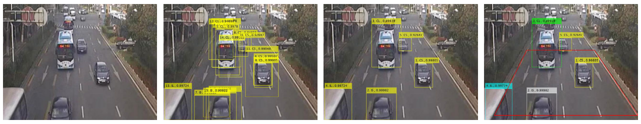
So far, four modified results ($box\text{-}2_k^{detector}$) of the selected test image have been obtained. Shown in Fig. 8, there were a total of 16 alternative objects in these four remove results, whereas there were only five correct objects (two complete small vehicles, one incomplete small vehicle, one complete large vehicle and one incomplete large vehicle) that really need to be detected. In this section, we will explain how to select the optimal results from these 16 alternative objects.

In order to better analyze and filter the alternative objects, a new variable was created, which is called $box\text{-}Alt_k$. We incorporated all of the alternative objects into this variable, which could be expressed in (17), as shown at the bottom of the next page, where n_1, n_2, n_3, n_4 represent the total number of object in $box\text{-}2_k^{Static\text{-}100}$, $box\text{-}2_k^{Static\text{-}200}$, $box\text{-}2_k^{Motion\text{-}100}$ and $box\text{-}2_k^{Motion\text{-}200}$ respectively, and the $box\text{-}Alt_k$ of the selected test frame is shown in Fig. 9b.

2) INTRODUCTION OF TWO IMPORTANT DISCRIMINANT METRICS

Before making the selection, two important metrics that describe the relationship of objects need to be elaborated. The first discriminant index is the description of the position relation of the center point of bounding box. In this study, that

$$\begin{aligned} k_{size,class,lane} &= \frac{size_{class,lane}^{near} - size_{class,lane}^{far}}{{y_{class,lane}^{lower\text{-}mid}}^{near} - {y_{class,lane}^{lower\text{-}mid}}^{far}} \\ b_{size,class,lane} &= \frac{size_{class,lane}^{far} \times {y_{class,lane}^{lower\text{-}mid}}^{near} - size_{class,lane}^{near} \times {y_{class,lane}^{lower\text{-}mid}}^{far}}{{y_{class,lane}^{lower\text{-}mid}}^{near} - {y_{class,lane}^{lower\text{-}mid}}^{far}} \end{aligned} \tag{8}$$



(a) (b) (c) (d)

FIGURE 9. Alternative objects and final selection of the selected frame. (a) I_k^{RGB}. (b) *box-Alt*$_k$. (c) *box-final*$_k$. (d) *box-final*$_k$ marked in different colors.

box_i is contained by box_j was defined as:

$$contained_i^{i\&j} = \begin{cases} 1, & if\ \{x_j^{left} \le x_i^{center} \le x_j^{right}\} \wedge \{y_j^{left} \le y_i^{center} \le y_j^{right}\} \\ 0, & otherwise \end{cases} \tag{18}$$

where

$$\begin{aligned} x_j^{left} &= x_j^{upper\text{-}left}, \quad y_j^{left} = x_j^{upper\text{-}left} \\ x_i^{center} &= x_i^{upper\text{-}left} + \frac{1}{2} \times width_i, \\ y_i^{center} &= y_i^{upper\text{-}left} + \frac{1}{2} \times height_i \\ x_j^{right} &= x_j^{upper\text{-}left} + width_j, \\ y_j^{right} &= y_j^{upper\text{-}left} + height_j \end{aligned} \tag{19}$$

The second metric is to examine the coincidence relationship between the two objects, which was used to describe the proportion of coinciding area to its own area. The proportion of box_i to its own area could be expressed as:

$$coinciding_i^{i\&j} = \frac{width^{i\&j} \times height^{i\&j}}{width_i \times height_i} \times 100\% \tag{20}$$

where

$$\begin{aligned} width^{i\&j} &= min(x_i^{right}, x_j^{right}) - min(x_i^{left}, x_j^{left}) \\ height^{i\&j} &= min(y_i^{right}, y_j^{right}) - min(y_i^{left}, y_j^{left}) \end{aligned} \tag{21}$$

3) IDENTIFICATION OF DETECTION ON SAME VEHICLE

Obviously, if two objects are the detected results of a same vehicle, they must overlap to some extent. When box_i and box_j meet one of the following conditions, we consider box_i and box_j overlap:

a. box_i is contained by box_j, or box_j is contained by box_i;
b. The proportion of box_i or box_j is over a certain threshold.

Here we use $pair^{i\&j}$ to represent the class combination of box_i and box_j, which could be expressed as:

$$pair^{i\&j} = \begin{cases} 1, & if\ class_i = class_j \\ 2, & if\ (class_i = CS, class_j = IS)\ or \\ & \quad (class_i = IS, class_j = CS) \\ 3, & if\ (class_i = CL, class_j = IL)\ or \\ & \quad (class_i = IL, class_j = CL) \\ 4, & otherwise \end{cases} \tag{22}$$

In this case, whether box_i and box_j overlap could be expressed as:

$$overlap^{i\&j} = \begin{cases} 1, & if\ (contained_i^{i\&j} + contained_j^{i\&j} \ge 1) \\ & \quad \vee (coinciding_i^{i\&j} \ge T_{pair^{i\&j}}) \\ & \quad \vee (coinciding_j^{i\&j} \ge T_{pair^{i\&j}}) \\ 0, & otherwise \end{cases} \tag{23}$$

where $T_{pair^{i\&j}}$ is the threshold of $contained\ ^{i\&j}$.

Therefore, if box_i and box_j are the detection on a same vehicle, they must satisfy $overlap^{i\&j} = 1$. It is worth noting that if box_j and box_l overlap, box_i and box_l should also overlap, which could be expressed as:

$$overlap^{i\&l} = \begin{cases} 1, & if\ (overlap^{i\&j} = 1) \wedge (overlap^{j\&l} = 1) \\ 0, & otherwise \end{cases} \tag{24}$$

$$\begin{aligned} & box\text{-}Alt_k \\ &= \left[\, box\text{-}2_k^{Static\text{-}100} \quad box\text{-}2_k^{Static\text{-}200} \quad box\text{-}2_k^{Motion\text{-}100} \quad box\text{-}2_k^{Motion\text{-}200} \,\right]^{\mathrm{T}} \\ &= \begin{bmatrix} box\text{-}Alt_{k\ 1} \\ \vdots \\ box\text{-}Alt_{k\ n_1} \\ \vdots \\ box\text{-}Alt_{k\ n_1+n_2} \\ \vdots \\ box\text{-}Alt_{k\ n_1+n_2+n_3} \\ \vdots \\ box\text{-}Alt_{k\ n_1+n_2+n_3+n_4} \end{bmatrix} \end{aligned} \tag{17}$$

For an image of Frame k, the $overlap^{i\&j}$ of each box_i and box_j could be integrated as a matrix $Overlap_k$:

$$Overlap_k(i,j) = overlap^{i\&j} \tag{25}$$

where $i, j = 1, 2, \ldots, n_1 + n_2 + n_3 + n_4$. In the meanwhile, the square matrix could be reduced to a simpler matrix according to (24). The algorithm for simplifying $Overlap_k$ was displayed in Table 5.

TABLE 5. The algorithm for simplifying $Overlap_k$.

```
Algorithm  Simplify $Overlap_k$ to $Overlap_k^{Sim}$
$Overlap_k^{Sim} = Overlap_k$;
for $i, j, l = 1$ to $n_1 + n_2 + n_3 + n_4$
    $max = max[i, j, l]$;
    $median = median[i, j, l]$;
    $min = min[i, j, l]$;
    if
      $Overlap_k^{Sim}(i,j) + Overlap_k^{Sim}(i,l) + Overlap_k^{Sim}(j,l) > 1$
    then
        $Overlap_k^{Sim}(max, max/median/min) = 0$;
        $Overlap_k^{Sim}(median, max/median/min) = 0$;
        $Overlap_k^{Sim}(min, max/median/min) = 1$;
    end
end
```

Moreover, if a row of $Overlap_k^{Sim}$ is all zero, the row will be deleted. In this case, $Overlap_k^{Sim}$ could be simplified to a matrix with a dimension of:

$$N^{final} \times (n_1 + n_2 + n_3 + n_4) \tag{26}$$

where N^{final} is the final number of object that should be detected in Frame k. Therefore, for a row, each element with a value of 1 constitutes an alternative set of a final selected object.

4) FINAL SELECTION FROM ALTERNATIVE OBJECTS

As mentioned above, each alternative object obtained after Section III-B is correct. Therefore, for each row in $Overlap_k^{Sim}$, the object with the highest confidence was selected as a final selected object, which could be expressed as:

$$box\text{-}final_k = \begin{bmatrix} box_1 & \cdots & box_i & \cdots & box_{N^{final}} \end{bmatrix}^{\mathrm{T}} \tag{27}$$

where

$$\begin{gathered} box_i = box\text{-}Alt_{k\ max^i}, \\ max^i \in \{max^i \mid score_{max^i} = max[score_l],\ Overlap_k^{Sim}(i,l) = 1\} \\ i = 1, \ldots, N^{final}, \quad l = 1, \ldots, n_1 + n_2 + n_3 + n_4 \end{gathered} \tag{28}$$

Take the selected frame as an example for illustration. The $Overlap_k$ is a square matrix with a dimension of 16 ($n_1 + n_2 + n_3 + n_4 = 16$), and it could be simplified to a 16×5 matrix using the algorithm introduced in Table 5, which means that the final number of detected objects in the selected frame is five ($N^{final} = 5$). To better illustrate the problem, we annotated i, $class_i$ and $score_i$ of each alternative object if $Overlap_k^{Sim}(i,j) = 1$, and displayed its tabular form in Table 6.

The object with the highest confidence $box\text{-}Alt_{k\ max^i}$ was marked in **bold** in each column. In this case, the final selection from the 16 alternative objects in (27), as shown at the bottom of the next page, shown in Fig. 9c. Moreover, to show the results more clearly, different classes were displayed in different colors, shown in Fig. 9d.

IV. EXPERIMENTAL RESULTS

A. EVALUATION INDEX

In this paper, five standardized evaluation indexes were chosen to evaluate the experimental results, that is, *Acc*, *Recall*, *Precision*, *F-measure* and *Kappa*.

The first metric *Acc* [48] is the overall accuracy which is the proportion of successfully detected and correctly classified frames in one video:

$$Acc = \frac{Correctly\ Detected\ and\ Classified\ Frame\ No.}{Testing\ Frame\ No.} \tag{30}$$

Recall is the percentage of successfully detected and correctly classified objects in all relevant objects, whereas *Precision* is the percentage of successfully detected and correctly classified objects in all detected objects. *F-measure* is the weighted harmonic average of *Recall* and *Precision*, which is a comprehensive evaluation of these two metrics. Here we make the weights equal to each other, and the three metrics could be expressed as [45]:

$$Recall = \frac{TP}{TP + FN}, \quad Precision = \frac{TP}{TP + FP} \tag{31}$$

$$F\text{-}measure = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} \tag{32}$$

where *TP* is the number of vehicles successfully detected and correctly classified, *FN* is the number of missed vehicles, and *FP* is the number of false objects detected as vehicles or misclassified vehicles. In this case, the average performance of four classes for *Recall*, *Precision* and *F-measure* in one video could be expressed as:

$$mRe = \frac{\sum Recall_{class}}{4} \tag{33}$$

$$mPr = \frac{\sum Recall_{class}}{4} \tag{34}$$

$$mF1 = \frac{\sum F\text{-}measure_{class}}{4} \tag{35}$$

The last metric *Kappa* is a measure that expresses the agreement between two annotators. The Cohen Kappa Score is defined as [46]:

$$Kappa = \frac{Acc - Pe}{1 - Pe} \tag{36}$$

where *Acc* is the accuracy (30) and *Pe* is the probability of agreement. *Kappa* fluctuates in the $[-1, 1]$, where $Kappa = 1$ means that both annotators are in complete agreement, whereas $Kappa \leq 0$ means no agreement at all.

TABLE 6. The tabular form of $Overlap_k^{Sim}$.

1.CS 0.9666	0	0	4.CS 0.9951	0	0	0	**8.CS 0.9981**	0	0	11.CS 0.9905	0	0	0	0	0
0	2.IS 0.9827	0	0	0	0	7.IS 0.9812	0	**9.IS 0.9998**	0	0	0	0	0	15.IS 0.9092	0
0	0	3.CL 0.9866	0	5.CL 0.9978	0	0	0	0	**10.CL 0.9988**	0	12.CL 0.9470	0	14.CL 0.9957	0	0
0	0	0	0	0	6.CS 0.8482	0	0	0	0	0	0	0	0	0	**16.CS 0.9265**
0	0	0	0	0	0	0	0	0	0	0	0	**13.IL 0.9972**	0	0	0

TABLE 7. The characteristics of each video.

Sequence	Road Environments			Traffic Conditions		Frame No.	
	Location	Weather	Shooting Angle	Traffic Volume (v/h)	Large Vehicle (%)	Training Set	Test Set
Video I	Urban I	Cloudy	Front	Medium (1,080)	High (17.78%)	100+200	8,700
Video II	Re-Urban I	Overcast	Front	Low (696)	High (10.34%)	0	9,000
Video III	Urban II	Overcast	Front	Medium (1,104)	High (13.04%)	0	9,000
Video IV	Express I	Cloudy	Side & Front	Medium (900)	Medium (2.67%)	0	9,000
Video V	Express II	Sunny	Side	High (3,336)	Medium (3.24%)	0	9,000
M-30-HD	High I	Cloudy	Rear	Medium (1,541)	Low (0.75%)	0	9,390
M-30	High I	Sunny	Rear & Side	High (3,547)	Medium (5.26%)	0	7,520

B. EXPERIMENTAL DATASET

For vehicle detection and classification based on traffic surveillance video, the road environments and traffic conditions have a great impact on the detection results. In terms of road types, the traffic environment of urban roads is more complex, so in general, urban roads bring more disadvantages than expressways and highways. From the perspective of weather, cloudy weather tends to bring about the impact of sudden illumination changes in video images, whereas the shadow of vehicles also affects the detection results under sunny conditions. Different shooting angles could also have a certain influence. For example, in the video taken from the side of the vehicle, occlusion phenomenon is relatively more common. Traffic conditions are also an important factor. Generally speaking, the higher the traffic flow and the higher the proportion of large vehicles, the more difficult the vehicle detection and classification will be.

Taking the above factors into consideration, except for the Video I which was used to train the four detectors (*Static*-100, *Static*-200, *Motion*-100 and *Motion*-200), six more real traffic videos were selected for experiments. Video II and Video I were recorded on the same road (Urban I), but Video II was shot after Urban I was rebuilt (Re-Urban I), whereas Video III was recorded on another urban road (Urban II) with a higher traffic volume. Video IV and Video V were recorded on two expressways in different weather conditions (Express I and Express II) and the vehicles were shot from the side. Video I, Video II, Video III, Video IV and Video V were all five-minute videos with 9,000 frames, and were recorded by ourselves in Nanjing, Jiangsu, China, which could be obtained from [40]. To further verify the robustness of proposed method, two videos taken from the rear of vehicle on a highway (High I) in different weather conditions were selected for experiment. They are two benchmark datasets called M-30-HD and M-30, which could be obtained from Road-Traffic Monitoring dataset [47]. More details of the characteristics of each video are shown in Table 7. At the same time, the actual vehicle number in each video and distribution of the fours classes were also listed in Table 8.

It should be noted that the 300 frames in the training sets (Training-100 and Training-200) were all selected from Video I. In order to make the experimental results more objective, these 300 frames were removed from the test set of Video I, and the size of its test set was reduced to 8,700. More details about the training set and test set are shown in Table 7.

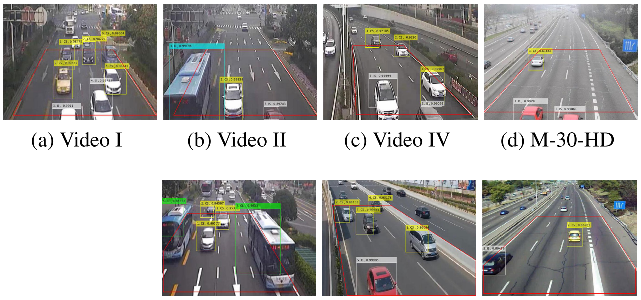
C. EXPERIMENTAL RESULTS AND ANALYSIS

Four detectors (*Static*-100, *Static*-200, *Motion*-100 and *Motion*-200) obtained by training with Video I were used to conduct experiments on the seven videos. We selected one frame from each of the seven videos as examples. As shown in Fig. 10, the vehicles within the detection area were detected and classified successfully. The robustness of proposed method could be verified by applying it to different videos.

$$\begin{aligned} box\text{-}final_k &= \begin{bmatrix} box_1 & box_2 & box_3 & box_4 & box_5 \end{bmatrix}^{\mathrm{T}} \\ &= \begin{bmatrix} box\text{-}Alt_{k\ 8} & box\text{-}Alt_{k\ 9} & box\text{-}Alt_{k\ 10} & box\text{-}Alt_{k\ 13} & box\text{-}Alt_{k\ 16} \end{bmatrix}^{\mathrm{T}}, \end{aligned} \tag{29}$$

TABLE 8. The class distribution in each video.

Sequence	Object No. in Test Set	Proportion of Each Class			
		CS	IS	CL	IL
Video I	12,016	65.63%	11.87%	15.13%	7.37%
Video II	5,140	74.90%	12.39%	8.44%	4.26%
Video III	9,618	71.73%	11.50%	11.38%	5.39%
Video IV	5,119	81.34%	15.14%	2.46%	1.05%
Video V	16,645	84.49%	14.65%	0.58%	0.28%
M-30-HD	13,005	85.64%	13.59%	0.61%	0.17%
M-30	12,855	80.05%	14.57%	4.01%	1.37%
Total	**74,398**	**78.35%**	**13.47%**	**5.60%**	**2.58%**



(a) Video I (b) Video II (c) Video IV (d) M-30-HD

(e) Video III (f) Video V (g) M-30

FIGURE 10. Examples of experimental results in each video.

TABLE 9. The experimental results of video I.

$class$	TP	FN	FP	$Recall$	$Precision$	$F\text{-}measure$
CS	7,532	340	47	0.9568	0.9938	0.9750
IS	1,287	122	47	0.9135	0.9651	0.9386
CL	1,780	28	39	0.9845	0.9786	0.9815
IL	858	6	20	0.9927	0.9770	0.9848

TABLE 10. The experimental results of video II.

$class$	TP	FN	FP	$Recall$	$Precision$	$F\text{-}measure$
CS	3,711	135	17	0.9650	0.9955	0.9800
IS	573	55	13	0.9127	0.9782	0.9443
CL	427	2	10	0.9953	0.9764	0.9858
IL	211	3	7	0.9860	0.9666	0.9762

TABLE 11. The experimental results of video III.

$class$	TP	FN	FP	$Recall$	$Precision$	$F\text{-}measure$
CS	6,552	336	74	0.9513	0.9888	0.9697
IS	993	89	42	0.9174	0.9596	0.9380
CL	1,046	27	31	0.9752	0.9713	0.9733
IL	488	9	28	0.9810	0.9457	0.9630

The experimental results in each video are shown in Table 9–Table 16.

In terms of the overall results, Video I and Video II achieved the highest accuracy and it is easy to explain. The training sets of the four detectors were selected from Video I, whereas Video I and Video II were recorded on a same urban road, but in different time periods (Urban I and Re-Urban I).

TABLE 12. The experimental results of video IV.

$class$	TP	FN	FP	$Recall$	$Precision$	$F\text{-}measure$
CS	3,966	192	11	0.9539	0.9973	0.9751
IS	706	62	12	0.9188	0.9836	0.9501
CL	119	0	7	0.9965	0.9450	0.9701
IL	50	1	3	0.9800	0.9452	0.9623

TABLE 13. The experimental results of video V.

$class$	TP	FN	FP	$Recall$	$Precision$	$F\text{-}measure$
CS	13,396	659	20	0.9531	0.9985	0.9753
IS	2,215	203	14	0.9162	0.9937	0.9534
CL	89	2	7	0.9806	0.9246	0.9518
IL	43	0	2	0.9926	0.9602	0.9761

TABLE 14. The experimental results of M-HD-30.

$class$	TP	FN	FP	$Recall$	$Precision$	$F\text{-}measure$
CS	10,317	819	25	0.9265	0.9976	0.9607
IS	1,591	174	6	0.9013	0.9964	0.9465
CL	76	3	5	0.9600	0.9405	0.9501
IL	21	1	6	0.9600	0.7875	0.8652

TABLE 15. The experimental results of M-30.

$class$	TP	FN	FP	$Recall$	$Precision$	$F\text{-}measure$
CS	9,931	359	13	0.9651	0.9986	0.9816
IS	1,756	114	7	0.9390	0.9960	0.9667
CL	501	9	5	0.9824	0.9899	0.9861
IL	165	7	4	0.9593	0.9763	0.9677

TABLE 16. The metric results in each video.

	Acc	mRe	mPr	$mF1$	$Kappa$
Video I	0.9869	0.9619	0.9786	0.9700	0.9755
Video II	0.9905	0.9647	0.9792	0.9716	0.9772
Video III	0.9831	0.9562	0.9664	0.9610	0.9636
Video IV	0.9889	0.9623	0.9677	0.9644	0.9651
Video V	0.9904	0.9606	0.9693	0.9641	0.9642
M-30-HD	0.9933	0.9369	0.9305	0.9306	0.9736
M-30	0.9916	0.9230	0.9773	0.9492	0.9756
Average	**0.9892**	**0.9522**	**0.9670**	**0.9587**	**0.9707**

However, as can be seen from Table 16, Video IV and M-30 also showed relatively good performance on some of the evaluation metrics. It is mainly due to the lower ratio of large vehicle, which reduces the frequency of errors such as vehicle occlusion. This can also be seen in the results of Video III. Although just like in Video I and Video II, they were both shooting from the front, but the results of Video III were relatively poorer than Video IV (from side) and M-30 (from rear), which is also due to the higher ratio of large vehicle.

For Video IV and Video V, overall, the detection accuracy of these two videos ranked the third and the fourth, higher than that of Video III taken from the front. Moreover, Video IV and Video V were shot on a cloudy day and a sunny day respectively, but the accuracy of results did not differ much. From these two points, it can be seen that the detection results are more influenced by the proportion of large vehicles, rather than the vehicle appearance and shooting angle, which further reflects the robustness and wide application range of the proposed method.

As for M-30-HD and M-30 shot from the rear, the *mRe* was relatively low. This is mainly due to the fact that the two static detectors trained with the front of vehicle did not "recognize" the rear very well.

Whereas for each specific *class*. According to Table 9–Table 15, *CL* and *CS* have better performance than *IS* and *IL*, which proves that the detectors have a better ability to identify the complete vehicle. This is mainly due to the higher percentage of complete vehicles in training sets.

In general, the average results of *Acc*, *mRe*, *mPr*, *mF*1 and *Kappa* were 0.9892, 0.9522, 0.9670, 0.9587 and 0.9707 respectively, which reached more than 0.95.

V. DISCUSSION

A. THE ADVANTAGES OF PROPOSED METHOD

1) SMALL SIZE OF TRAINING SET

The sizes of training set (Training-100 and Training-200) used for training detectors were small, which are 100 images and 200 images respectively. In this case, the difficulty of a lot of manual labeling could be avoided and the training time could be reduced greatly. Moreover, a small amount of image learning can reduce the overfitting of detector, so that the application range could be increased.

2) GOOD COMPLEMENTARITY OF DIFFERENT DETECTORS BASED ON STATIC AND MOTION FEATURES

The four detectors (*Static*-100, *Static*-200, *Motion*-100 and *Motion*-200) for experiments have different performances, which makes the detectors complementary to each other.

Take Video I as an example. As shown in Table 17, the *mPr* of static detectors (*Static*-100 and *Static*-200) is significantly lower than motion detectors (*Motion*-100 and *Motion*-200), whereas the *mRe* is relatively higher. This illustrates that the error of missing detection are more common for the motion detectors, whereas the errors of detecting redundantly are more common for the static detectors. From the perspective of the number of training sets, compared with the detectors trained with Training-100, the detectors trained with Training-200 has a relatively lower probability of missing detection, but the error frequency of detecting redundantly also has a certain increase, no matter for static detectors or motion detectors.

3) EFFECTIVENESS OF VEHICLE CLASSIFICATION FORMS

The vehicles were divided into four classes in this study, namely complete small vehicle (*CS*), incomplete small vehicle (*IS*), complete large vehicle (*CL*) and incomplete large vehicle (*IL*). However, the purpose of such classification is not to distinguish complete vehicles from incomplete vehicles, but to improve the detection capability.

To better illustrate this point, we did a comparison experiment. We divided the vehicles into two classes (small vehicle and large vehicle) and trained them with the same training set. As shown in Table 17, no matter for the *mRe*, *mPr* or *mF*1, the detection results based on four classes are significantly improved compared with those based on two classes, which proved the validity of this classification method.

TABLE 17. Experimental results of video I with different detectors.

	Class No.	TP	FN	FP	mRe	mPr	$mF1$
$Static$-100	4	11,058	633	2,320	0.9361	0.7325	0.8177
$Static$-200	4	11,191	440	2,879	0.9515	0.6918	0.7952
$Motion$-100	4	9,373	2,442	270	0.6784	0.9445	0.7839
$Motion$-200	4	9,534	2,165	547	0.7375	0.9025	0.8070
Final Detection	2	10,697	1,278	158	0.9163	0.9805	0.9468
Final Detection	**4**	**11,833**	**142**	**56**	**0.9907**	**0.9938**	**0.9922**

4) WIDE APPLICABILITY AND ROBUSTNESS OF PROPOSED METHOD

The four detectors used in the experiments were all trained with Video I, which is a video of vehicle being shot from the front on an urban road. To verify the wide applicability of proposed method, six more traffic videos with different characteristics were selected for experiments. For the location, there were two more videos taken on urban road, and two on expressway, two on highway. Considering the adverse effects of sudden illumination changes on cloudy days and shadows on sunny days, videos taken on cloudy and sunny days were also studied. For the shooting angle, vehicles shot from front, side and rear were all considered in experiments. Whereas the traffic conditions (traffic flow and proportion of large vehicles) are also in different levels for different videos.

The algorithms for remove, selection and reorganization of detected objects were designed to realize the vehicle detection with high robustness. In this framework, some obvious errors will be removed first based on the detection area, the size of bounding box, the motion features and the detected confidence. The reorganization of alternative detection from the four different detectors greatly reduces the probability of missing error. As shown in Table 17, although the detection accuracy of the four detectors (*Static*-100, *Static*-200, *Motion*-100 and *Motion*-200) is relatively low, thanks to the different performance of detectors and the effectiveness of remove, selection and reorganization, the accuracy of the final detection has been greatly improved.

In general, the experiment results of *Acc*, *mRe*, *mPr*, *mF*1 and *Kappa* for each video all reached more than 0.92 as shown in Table 16.

B. COMPARISON WITH RECENT STATE-OF-THE-ART METHODS

Comparing with recent state-of-the-art methods, the results of accuracy and speed are shown in Table 18. In general, the construction of model and backbone, selection

TABLE 18. Comparison of experimental results with recent state-of-the-art methods.

	Model	Backbone	Training Set	mPr	FPS
Cao *et al.* [49]	SSD	ResNet	5,000	0.9035	20
Dai *et al.* [50]	SSD	VGG16	6,134	0.8910	56
Sang *et al.* [51]	YOLOv2	DarkNet19	8,680	0.9478	26
Mao *et al.* [52]	YOLOv3	DarkNet53	13,179	0.9022	40
Dai *et al.* [50]	Faster-RCNN	VGG16	6,134	0.9010	7
Proposed Method	**Faster-RCNN**	**AlexNet**	**100 and 200**	**0.9670**	**21**

and size of training set, setting of parameter or other aspects could have different effects on the accuracy and time consuming of the results. At the same time, precision and speed are often difficult to reconcile. For example, the YOLOv2+DarkNet19 built by Sang *et al.* [51] was more accurate than the others, but the speed is significantly lower than the SSD+VGG16 built by Dai *et al.* [50]. Moreover, the accuracy of the results may not necessarily improve with the increasing of training sets, and suitability may be more important, such as SSD+ResNet [49] and SSD+VGG16 [50], YOLOv2+DarkNet19 [51] and YOLOv3+DarkNet53 [52].

The network used in the proposed method is Faster-RCNN+AlexNet. Although the training sets of detectors used in the experiments were small, benefited from the different performance of different detectors and the effectiveness of remove, selection and reorganization, the accuracy of proposed method is at a relatively high level. Moreover, compared with Faster-RCNN+VGG16 [50], the selection of AlexNet may also contribute to the improvement of detection speed. In general, the method presented in this paper achieves a relative balance between accuracy and speed.

C. ANALYSIS FOR THE IMPROVEMENT OF PROPOSED METHOD

According to the analysis in Section IV-C, errors usually occur in two ways. The first is due to the complex traffic environment, that is, the large vehicle block the small one, resulting in the small one cannot appear in the image, which is also a defect for the detection with a single frame image. This kind of error is most often seen on complex urban roads with a high proportion of large vehicles, such as Video I and Video III. To think about that, the vehicle relationship between frames could be considered to reduce missing errors.

The second one is due to the detector failure of ''recognizing'' vehicle, which occurred most in M-30-HD and M-30. For one thing, it is because the too small vehicles at a distance in training set were not marked, resulting the failure of identifying the too small size of vehicles. For another thing, it is due to the failure of "recognizing" the rear of vehicle. Therefore, an appropriate amount of vehicles with smaller sizes and vehicles shot from the rear could be added to the training set to improve the detector capability.

VI. CONCLUSION

In this paper, we proposed a vehicle detection and classification method which can adapted to different roads and environments well. Two small training sets were used to train the four different detectors, which reduced the workload of data collection and manual annotation. In addition, training time could also be greatly reduced. Although the training process is relatively less complex, benefited from the different performance and good complementarity of the four detectors, as well as the algorithms for remove, selection and reorganization of detected objects, the application scope of this method could be expanded.

Experimental results showed that the proposed system performs well in different traffic videos with different characteristics, such as different shooting locations, weather conditions, shooting angles and traffic environments. The proposed method successfully detected and classified the vehicles with a high performance, and the overall result reached more than 0.95.

In future works, we intend to optimize the detection and classification algorithms by combining the vehicle relationship between frames to reduce missing errors. Also, an appropriate amount of vehicles with smaller sizes and vehicles shot from the rear could be added to the training set to improve the detector capability.

REFERENCES

[1] L.-W. Tsai, J.-W. Hsieh, and K.-C. Fan, ''Vehicle detection using normalized color and edge map,'' *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 850–864, Mar. 2007.

[2] T. Celik and H. Kusetogullari, ''Solar-powered automated road surveillance system for speed violation detection,'' *IEEE Trans. Ind. Electron.*, vol. 57, no. 9, pp. 3216–3227, Sep. 2010.

[3] Z. Yang and L. S. C. Pun-Cheng, ''Vehicle detection in intelligent transportation systems and its applications under varying environments: A review,'' *Image Vis. Comput.*, vol. 69, pp. 143–154, Jan. 2018.

[4] S. Sivaraman and M. M. Trivedi, ''Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1773–1795, Dec. 2013.

[5] W. Enkelmann, ''Obstacle detection by evaluation of optical flow fields from image sequences,'' *Image Vis. Comput.*, vol. 9, no. 3, pp. 160–168, Jun. 1991.

[6] B. K. Horn and B. G. Schunck, ''Determining optical flow,'' *Artif. Intell.*, vol. 17, nos. 1–3, pp. 185–203, 1981.

[7] W. Zhan and X. Ji, ''Algorithm research on moving vehicles detection,'' in *Proc. Int. Conf. Adv. Control Eng. Inf. Sci.*, Yunnam, China, vol. 15, Aug. 2011, pp. 5483–5487.

[8] J. B. Kim and H. J. Kim, ''Efficient region-based motion segmentation for a video monitoring system,'' *Pattern Recognit. Lett.*, vol. 24, nos. 1–3, pp. 113–128, Jan. 2003.

[9] S. Kul, S. Eken, and A. Sayar, ''Distributed and collaborative real-time vehicle detection and classification over the video streams,'' *Int. J. Adv. Robot. Syst.*, vol. 14, no. 4, pp. 1–12, 2017.

[10] M. Bertozzi, A. Broggi, and S. Castelluccio, ''A real-time oriented system for vehicle detection,'' *J. Syst. Archit.*, vol. 43, nos. 1–5, pp. 317–325, Mar. 1997.

[11] R. M. Haralick, K. Shanmugam, and I. Dinstein, ''Textural features for image classification,'' *IEEE Trans. Syst., Man, Cybern.*, vols. SMC-3, no. 6, pp. 610–621, Nov. 1973.

[12] N. D. Matthews, P. E. An, D. Charnley, and C. J. Harris, ''Vehicle detection and recognition in greyscale imagery,'' *Control Eng. Pract.*, vol. 4, no. 4, pp. 473–479, Apr. 1996.

[13] Y. Yang, X. Gao, and G. Yang, ''Study the method of vehicle license locating based on color segmentation,'' in *Proc. Int. Conf. Adv. Control Eng. Inf. Sci.*, Yunnam, China, Aug. 2011, vol. 15, pp. 1324–1329.

[14] J. Yang, Y. Wang, A. Sowmya, and Z. Li, ''Vehicle detection and tracking with low-angle cameras,'' in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 685–688.

[15] D.-Y. Chen, Y.-H. Lin, and Y.-J. Peng, "Nighttime brake-light detection by nakagami imaging," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1627–1637, Dec. 2012.

[16] V. Abolghasemi and A. Ahmadyfard, "An edge-based color-aided method for license plate detection," *Image Vis. Comput.*, vol. 27, no. 8, pp. 1134–1142, Jul. 2009.

[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, USA, 2005, pp. 886–893.

[18] C. P. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *Proc. 6th Int. Conf. Comput. Vis.*, Mumbai, India, 1998, pp. 555–562. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=710772&isnumber=15374, doi: 10.1109/ICCV.1998.710772.

[19] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Kauai, HI, USA, Dec. 2001, pp. I511–I518.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. 26th Annu. Conf. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, Dec. 2012, vol. 2, pp. 1097–1105.

[21] M. Yang, B. Li, H. Fan, and Y. Jiang, "Randomized spatial pooling in deep convolutional networks for scene recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Zurich, Switzerland, Sep. 2015, pp. 346–361.

[22] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 580–587.

[23] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1440–1448.

[24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[25] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, 2017, pp. 2980–2988.

[26] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 779–788.

[27] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)* (Lecture Notes in Computer Science), vol. 9905. Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 21–37.

[28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2999–3007.

[29] B. Wu, A. Wan, F. Iandola, P. H. Jin, and K. Keutzer, "SqueezeDet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, Jul. 2017, pp. 446–454.

[30] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *Proc. 15th Eur. Conf. Comput. Vis.*, Munich, Germany, Sep. 2018, pp. 765–781.

[31] Y. Zhou, Y. Pei, Z. Li, L. Fang, Y. Zhao, and W. Yi, "Vehicle weight identification system for spatiotemporal load distribution on bridges based on non-contact machine vision technology and deep learning algorithms," *Measurement*, vol. 159, Jul. 2020, Art. no. 107801.

[32] Y. Wu, M. Abdel-Aty, O. Zheng, Q. Cai, and S. Zhang, "Automated safety diagnosis based on unmanned aerial vehicle video and deep learning algorithm," *Transp. Res. Rec.*, vol. 2674, no. 8, pp. 350–359, Aug. 2020.

[33] K.-J. Kim, P.-K. Kim, Y.-S. Chung, and D.-H. Choi, "Multi-scale detector for accurate vehicle detection in traffic surveillance data," *IEEE Access*, vol. 7, pp. 78311–78319, 2019.

[34] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. 30th Int. Conf. Mach. Learn.*, Atlanta, GA, USA, Jun. 2013, pp. 2347–2355.

[35] N. Abramson, G. Sebestyen, and D. Braverman, "Pattern-recognition and machine learning," *IEEE Trans. Inf. Theory*, vol. IT-9, no. 4, p. 257, May 1963.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, San Diego, CA, USA, May 2015, pp. 1–5.

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[39] *Imagenet Large Scale Visual Recognition Competition 2012 (Ilsvrc2012)*. Accessed: May 28, 2020. [Online]. Available: http://image-net.org/challenges/LSVRC/2012/index

[40] *Videos for Experiments*. Accessed: Jun. 17, 2020. [Online]. Available: https://pan.baidu.com/s/152kinqMCrMp42MOILRwExQ

[41] C. Sun and S. G. Ritchie, "Heuristic vehicle classification using inductive signatures on freeways," *Transp. Res. Rec.*, vol. 17, pp. 130–136, Feb. 2000.

[42] *Highway Capacity Manual 2000*, Transportation Research Board, Washington, DC, USA, 2000.

[43] R. Johnson, "Accelerating stochastic gradient descent using predictive variance reduction," in *Proc. 27th Annu. Conf. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, Dec. 2013, pp. 315–323.

[44] B. Gloyer, H. K. Aghajan, K.-Y. Siu, and T. Kailath, "Video-based freeway-monitoring system using recursive vehicle tracking," in *Proc. Image Video Process. III*, San Jose, CA, USA, Mar. 1995, pp. 173–180.

[45] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "CDnet 2014: An expanded change detection benchmark dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Columbus, OH, USA, Jun. 2014, pp. 393–400.

[46] J. Cohen, "A coefficient of agreement for nominal scales," *Edu. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, 1960.

[47] *Gram Road-Traffic Monitoring*. Accessed: Jan. 25, 2020. [Online]. Available: http://agamenon.tsc.uah.es/Personales/rlopez/data/rtm/

[48] H. Jung, M.-K. Choi, J. Jung, J.-H. Lee, S. Kwon, and W. Y. Jung, "ResNet-based vehicle classification and localization in traffic surveillance systems," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, Jul. 2017, pp. 934–940.

[49] W. Cao, J. Yuan, Z. He, Z. Zhang, and Z. He, "Fast deep neural networks with knowledge guided training and predicted regions of interests for real-time video object detection," *IEEE Access*, vol. 6, pp. 8990–8999, 2018.

[50] Z. Dai, H. Song, X. Wang, Y. Fang, X. Yun, Z. Zhang, and H. Li, "Video-based vehicle counting framework," *IEEE Access*, vol. 7, pp. 64460–64470, 2019.

[51] J. Sang, Z. Wu, P. Guo, H. Hu, H. Xiang, Q. Zhang, and B. Cai, "An improved YOLOv2 for vehicle detection," *Sensors*, vol. 18, no. 12, p. 4272, Dec. 2018.

[52] Q.-C. Mao, H.-M. Sun, L.-Q. Zuo, and R.-S. Jia, "Finding every car: A traffic surveillance multi-scale vehicle object detection method," *Int. J. Speech Technol.*, vol. 50, no. 10, pp. 3125–3136, Oct. 2020.

YUE CHEN received the bachelor's degree in traffic engineering and the master's degree in transportation engineering from Southeast University, Nanjing, China, in 2015 and 2018, respectively, where she is currently pursuing the Ph.D. degree in transportation engineering.

Her current research interests include intelligent transportation systems, image processing and recognition, machine learning, and transportation planning and management.

WUSHENG HU received the Ph.D. degree from Hohai University, in 2001. He is currently a Professor with the Department of Surveying Engineering, School of Transportation, Southeast University, Nanjing, China.

His current research is primarily involved in deformation monitoring and analysis research, GPS engineering application and data processing, application research of neural network engineering and machine learning, and GIS technology development and application research.