Received: 31 January 2021 | Revised: 11 April 2021 | Accepted: 15 May 2021

IET Image Processing

DOI: 10.1049/ipr2.12295

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH PAPER

Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector

Wei Jia | Shiquan Xu | Zhen Liang | Yang Zhao | Hai Min | Shujie Li | Ye Yu

School of Computer Science and Information, Hefei University of Technology, Hefei, People's Republic of China

Correspondence
Ye Yu, School of Computer Science and Information, Hefei University of Technology, No.193 Tunxi Road, Hefei, People's Republic of China.
Email: yuye@hfut.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 62076086, 61673157, 61972129, 61972127; Key Research and Development Program in Anhui Province, Grant/Award Numbers: 202004d07020008, 201904d07020010

Abstract

In traffic accidents, motorcycle accidents are the main cause of casualties, especially in developing countries. The main cause of fatal injuries in motorcycle accidents is that motorcycle riders or passengers do not wear helmets. In this paper, an automatic helmet detection of motorcyclists method based on deep learning is presented. The method consists of two steps. The first step uses the improved YOLOv5 detector to detect motorcycles (including motorcyclists) from video surveillance. The second step takes the motorcycles detected in the previous step as input and continues to use the improved YOLOv5 detector to detect whether the motorcyclists wear helmets. The improvement of the YOLOv5 detector includes the fusion of triplet attention and the use of soft-NMS instead of NMS. A new motorcycle helmet dataset (HFUT-MH) is being proposed, which is larger and more comprehensive than the existing dataset derived from multiple traffic monitoring in Chinese cities. Finally, the proposed method is verified by experiments and compared with other state-of-the-art methods. Our method achieves mAP of 97.7%, F1-score of 92.7% and frames per second (FPS) of 63, which outperforms other state-of-the-art detection methods.

1 | INTRODUCTION

According to the latest report of 'Global Road Safety Status in 2018' released by the World Health Organization (WHO) [1], about 1.35 million people die in road traffic accidents every year, among which 28% die from motorcyclists. Especially in some underdeveloped areas, due to the restrictions of urban infrastructure and economic conditions, motorcycles have become the main tool of transportation, and the death rate of road traffic in these areas is about three times that in developed areas. In Southeast Asia and the Western Pacific region, such as India, Vietnam, Indonesia and other countries, motorcycle traffic accident deaths accounted for 43% and 36% of all traffic accident deaths respectively. The WHO points out that the head injury of motorcyclists is the main cause of death. If motorcyclists wear helmets correctly, the risk of death can be reduced by 42%, and the risk of head injury can be reduced by 69%. Therefore, it is very necessary for motorcyclists to wear helmets [1]. However, in some developing countries, the rate of wearing helmets has been very low for various reasons. For example, according to the data of Thailand road foundation [2], motorcycle traffic accidents in Thailand cause about 5500 deaths every year, and only 20% of passengers in the back seat of motorcycles wear helmets. In order to increase the use of helmets, the Indian government has proposed various penalties under the Motor Vehicles (Amendment) Law of 2019. Under section 194d, a motorcyclist who does not wear a helmet will be fined 1000 rupees and disqualified for the driving license for three months. Even if these laws exist, people will still try to escape the arrest of traffic police, and strict law enforcement also requires a lot of police, which is time-consuming and costly. Therefore, it is necessary to develop an automatic helmet detection of motorcyclists system based on deep learning to reduce the number of deaths in motorcycle traffic accidents. In recent years, with the rapid development of deep learning, convolutional neural network (CNN) has been widely used, such as semantic segmentation, object detection, all of which have made great breakthroughs. Semantic segmentation, a pixel-level vision task, is developed rapidly by using CNNs. Wang et al. [3] propose a weakly supervised adversarial domain adaptation to improve the

segmentation performance from synthetic data to real scenes. At present, object detection methods based on deep learning are generally divided into two categories: one-stage algorithms and two-stage algorithms. The two-stage algorithms are represented by the faster R-CNN [4] and the mask R-CNN [5]. These algorithms have high detection accuracy, but the speed is very slow, which makes it difficult to realize real-time detection. In order to solve this problem, researchers proposed one-stage algorithms, including you only look once (YOLO) [6], single shot detector (SSD) [7], RetinaNet [8] based on anchor and adaptive training sample selection (ATSS) [9], Fully Convolutional One-Stage Object Detection (FCOS) [10], RepPoints [11] based on anchor free. There are five versions of the YOLO algorithm. In this paper, the fifth generation YOLO detection algorithm is adopted, namely YOLOv5 [12]. It has both high speed and high precision, and the model parameters are very small. The smallest version is only 7.3M.

There are many research on visual detection of an intelligent transportation system, such as road detection [13], traffic flow prediction [14], license plate recognition [15] and vehicle recognition [16], but there are few research on helmet detection of motorcyclists. At present, although some motorcycle helmet detection methods have been proposed (refer to Section 2 for details), there are still some problems: (1) Most of the methods use traditional machine learning technology, which is poor in accuracy and speed. (2) For the detection of motorcycles, most of them use the background subtraction method to get the foreground objects, and then classify them to get motorcycles. However, in a crowded scene, when the motorcycles run slowly or stop, the detection will fail. (3) For helmet detection, many methods use a classification algorithm. However, when there is more than one persons on a motorcycle, it is difficult to judge whether someone is not wearing a helmet. (4) Lack of datasets for complex traffic monitoring scenarios.

The main contributions of this paper are as follows:

1. In this paper, we propose an automatic helmet detection of motorcyclists method using an improved YOLOv5 detector which integrates the triplet attention. The method consists of two stages: motorcycle detection and helmet detection, and can effectively improve the precision and recall of helmet detection.
2. We propose a large-scale motorcycle helmet dataset (HFUT-MH), which is obtained from traffic monitoring of many cities in China, including different illumination, different perspectives and different congestion levels.
3. Soft-NMS [17] post-processing is used to solve the occlusion problem in crowded scenes.
4. In order to demonstrate the effectiveness of our method, we have carried out evaluation experiments with other state-of-the-art detection methods.
5. The remainder of this paper is arranged as follows. Section 2 reviews related works. Section 3 focuses on the helmet detection of motorcyclists method we proposed. Section 4 introduces our motorcycle helmet dataset, HFUT-MH. Section 5 report experimental results and analysis. Section 6 summarizes this paper.

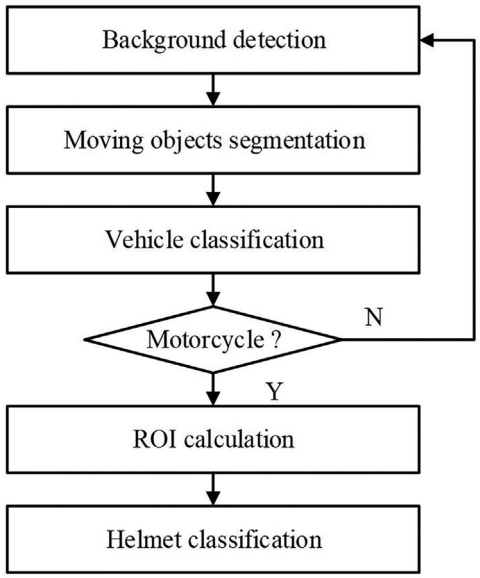

FIGURE 1 Motorcycle helmet detection flow based on the traditional method

2 | RELATED WORKS

The first step in the helmet detection of motorcyclists is usually motorcycle detection. In [18], the research status of motorcycle detection based on traditional methods and deep learning is reviewed in detail. Currently, some methods based on video or image have been proposed for helmet detection. It can be divided into two categories: The first one is based on traditional methods, and the second is based on deep learning. From the existing literature, it can be found that most of the current methods are traditional methods, and there are relatively few methods based on deep learning.

2.1 | Helmet detection based on traditional methods

Traditional methods [19–26] mostly adopt similar ideas as shown in Figure 1. The first step is moving object detection. First, the motion segmentation method is used to extract moving objects from surveillance videos. Common motion segmentation methods include optical flow, frame difference and background subtraction [27], [28]. Second, hand-designed feature descriptors, such as local binary pattern (LBP), histogram of oriented gradient (HOG), scale invariant feature transform (SIFT), are used to extract the features of motorcycles and other vehicles. Finally, motorcycles are classified by binary classifiers (such as support vector machine (SVM) and K-nearest neighbor (KNN)). Silva et al. [26] divided the problem of detecting helmet use by motorcyclists into two steps. The first step consists of the segment and classify the vehicle images. This step aims to detect all the moving objects in the scene. The second step is helmet detection, which uses a hybrid descriptor to extract image features and a support vector machine

classifier to classify an image in helmet and a non-helmet. Dahiya et al. [19] first detected bike riders from surveillance video using background subtraction and object segmentation. Then it determines whether the bike-rider is using a helmet or not using visual features and binary classifier. Talaulikar et al. [24] also applied a background subtraction technique to identify moving vehicles and principal component analysis (PCA) to the derived features.

The disadvantages of this kind of method are: (1) it is difficult to achieve real-time speed due to multi-stage operation; (2) when a motorcycle has more than one rider, especially when the person without a helmet is partially covered by the person with a helmet, it is difficult to judge whether there is a person without a helmet by using a classification algorithm; (3) when there is road congestion, camera jitter, branch jitter or other disturbances, the method based on motion segmentation will be greatly affected.

2.2 | Helmet detection based on deep learning

In recent years, researchers have proposed some methods based on deep learning. In [29], the background subtraction method and the SMO classifier are used to detect motorcycles from videos. Then, hand-crafted features and CNN are used to classify helmet and *no helmet* respectively. Finally, it is verified that the accuracy of CNN is higher than that of manual features. In [30], adaptive background subtraction is used to obtain the moving object on the video frame. Then, CNN is used to classify motorcyclists in moving objects. Finally, they continue to use CNN to classify the top quarter area of motorcycles to further identify that motorcyclists do not have helmets. In [31], Gaussian mixture model (GMM) is used to segment foreground objects, and then label them. Then, the system uses a faster region-based CNN (faster R-CNN) to detect motorcycles in the marked foreground objects to ensure the existence of motorcyclists. Later, the faster R-CNN was also used to detect motorcyclists with or without helmets. In [29], [30] and [31], although the helmet detection adopts the deep learning method, the traditional background subtraction is still used to obtain the foreground target in the motorcycle detection stage, which will be very poor in the crowded scene. In [32] and [33], it is proposed to use YOLOv3 [34] algorithm to detect whether a motorcyclist is wearing a helmet, but the detection of motorcycles is not reported. In [35] and [36], they first used the YOLOv3 algorithm to detect the motorcycle and person in the picture, and then calculated the overlapping area of the bounding box between the motorcycle and the person to determine the person on the motorcycle. Finally, they used

the YOLOv3 algorithm to detect whether the motorcyclist wore a helmet. However, in the view of traffic monitoring, motorcyclists and motorcycles are highly overlapping, so it is unnecessary to detect motorcyclists separately. In [37], [38] and [39], they proposed to use SSD or YOLOv3 algorithm to detect the motorcycle area, then extract the upper part of the image, and use the classification algorithm to identify the helmet and non-helmet. Similarly, when there is more than one person on the motorcycle, the classification algorithm will be invalid. In [40], [41] and [42], they regard the motorcycle and the motorcyclist as a whole, and then directly use the CNN model to detect whether the rider of the motorcycle is wearing a helmet. This one-step coarse-grained detection method has very low accuracy.

3 | METHODOLOGY

In urban traffic, there are many kinds of vehicles on the road, such as two-wheeled vehicles, three-wheeled vehicles, four-wheeled vehicles, and road congestion often occurs. In such a complex scene, it is a very challenging task to accurately detect the motorcycle and judge whether the rider on the motorcycle is wearing a helmet. Although a variety of helmet detection of motorcyclists methods have been proposed in some literature, these methods have many shortcomings, such as the limitations of accuracy and speed of traditional methods, and lack of high-quality traffic monitoring scene datasets. In this section, we propose a real-time and accurate automatic helmet detection of motorcyclists method based on deep learning, which includes two steps, as shown in Figure 2. The first step is motorcycle detection. First, the image to be detected is obtained from the video surveillance, and then the improved the YOLOv5 algorithm, namely YOLOv5-MD, is used to detect the riding motorcycle in the image, which contains at least one rider. The second step is helmet detection, which takes the motorcycle region detected in the first step as the input of the second step, and then continues to use the improved YOLOv5 algorithm, namely YOLOv5-HD, to detect whether the motorcyclists are not wearing helmets. Because the tasks of motorcycle detection and helmet detection are quite different, the network is designed for each stage, that is, YOLOv5-MD and YOLOv5-HD, in order to better improve the detection performance. Section 3.1 introduces the proposed network, Section 3.2 introduces the motorcycle detection method, and Section 3.3 introduces the helmet detection method.

3.1 | The proposed network

3.1.1 | YOLOv5 network

YOLO is a classical one-stage object detection algorithm. It turns the detection problem into a regression problem. Instead of extracting RoI, it directly generates the bounding box coordinates and probability of each class by the regression method. Compared with faster R-CNN, it greatly improves the detection speed. In 2020, the fifth version of YOLO was proposed by utralytics and named YOLOv5, which surpasses all previous versions in speed and accuracy. The YOLOv5 algorithm uses the parameters depth_multiple and width_multiple to adjust the width and depth of the backbone network, so as to get four versions of the model, which are YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x. YOLOv5s is the simplest version with

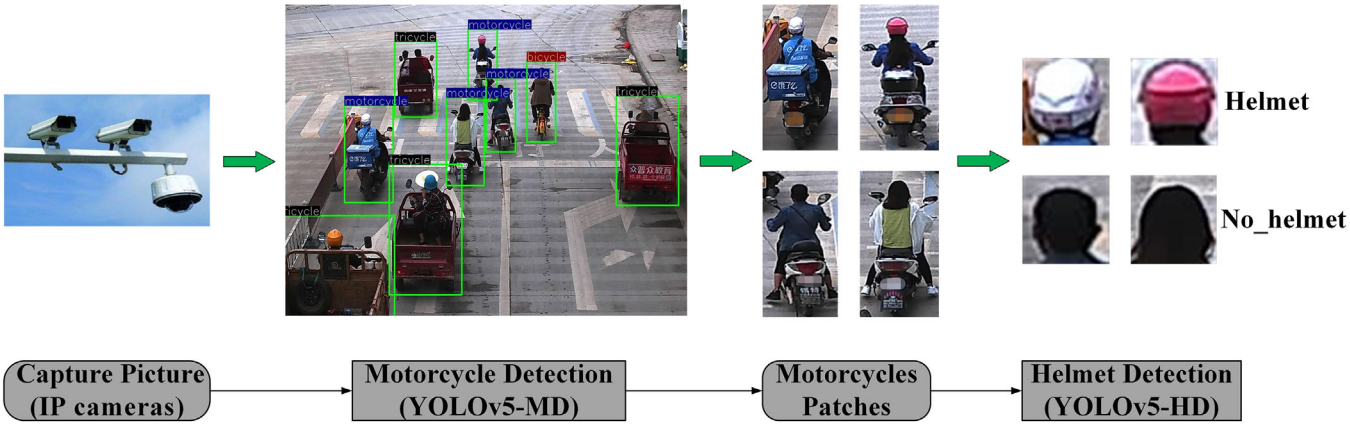

FIGURE 2 The overall framework of the proposed method

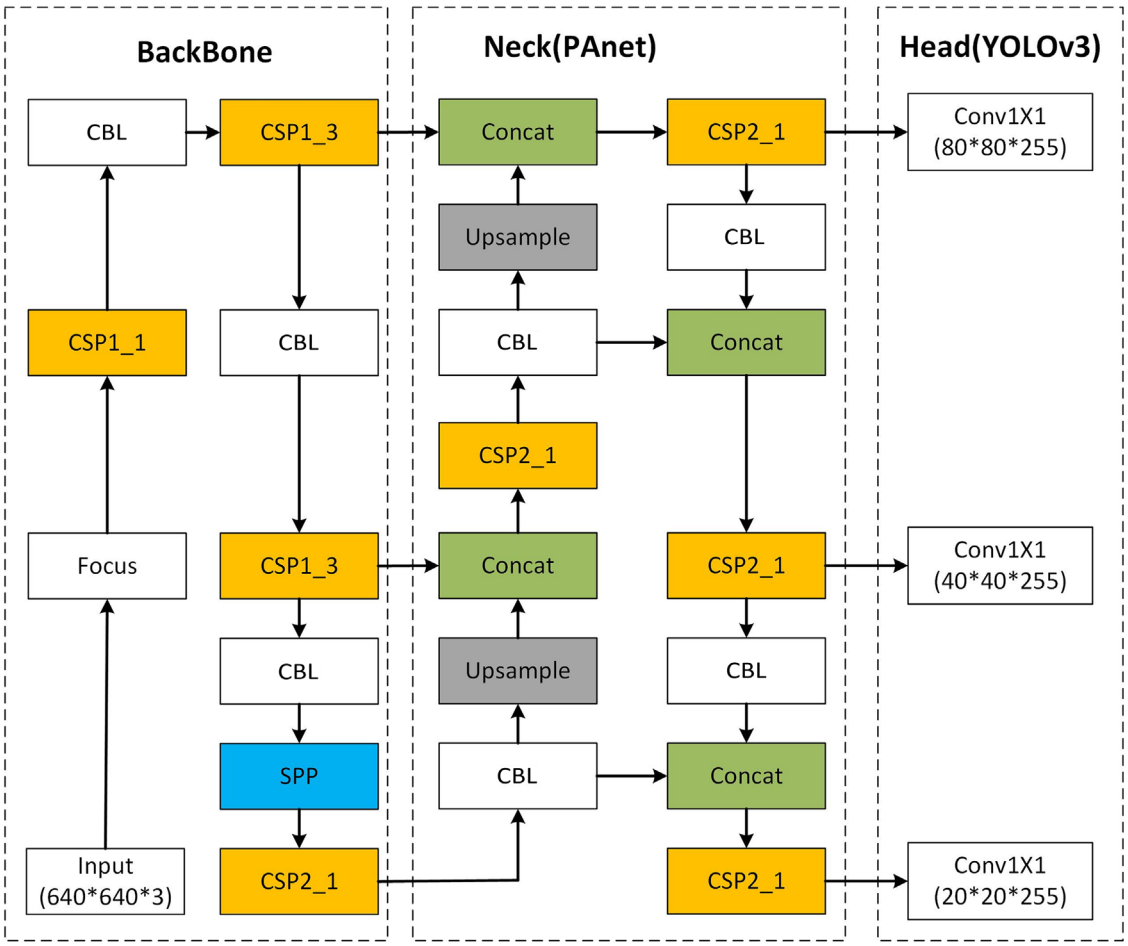

FIGURE 3 YOLOv5s network structure

the smallest model parameters and the fastest detection speed. Its network structure is shown in Figure 3, which is composed of focus, Conv-Bn-Leakyrelu (CBL) and CSP1_x, CSP2_x and spatial pyramid pooling (SPP) modules. The focus block mainly contains four parallel slice layers to tackle with the input image. The CBL block contains a convolutional layer, batch normalization layer and hard-wish function. CSP1_x block contains CBL blocks and x residual connection units. CSP2_x block contains x CBL blocks. SPP block mainly contains three maxpool layers.

YOLOv5s divides the model into three parts: the backbone network part, the feature enhancement part and the head part. Each part has different functions.

The backbone network part is used to extract image features. First, the focus structure is used to extract pixels from high-resolution images periodically and reconstruct them into low-resolution images. That is to say, the four adjacent positions of the images are stacked, and the information of WH dimension is focused into the C channel space to improve the receptive field of each point and reduce the loss of original information. The design of the module is mainly to reduce the amount of calculation and speed up. Then, drawing on the design idea of CSPNet [43], the CSP1_x and CSP2_x modules are designed. The module first divides the feature mapping of the basic layer into two parts, and then combines them through the cross-stage hierarchical structure, which reduces the amount of calculation and ensures accuracy. In the last part of the backbone network, using the SPP network, this module can further expand the receptive field and help to separate contextual features.

The feature enhancement part is used to further improve the feature extraction ability. It uses the idea of PANet [44] to design the structure of FPN+PAN. First, it uses the FPN structure to convey strong semantic features from top to bottom, and then uses the feature pyramid structure constructed by the PAN module to convey strong positioning features from bottom to top. Through this method, it is used to fuse features between different layers.

The head part inherits the head structure of YOLOv3, which has three branches. The prediction information includes object coordinates, category and confidence. The main improvement is to use complete intersection over union (CIOU) loss as the bounding box region loss.

3.1.2 | YOLOv5 with an attention mechanism

In motorcycle detection stage, due to the similar front appearance of motorcycles, tricycles and bicycles in riding state, the gap between the three different classes is relatively minor, while the differences between motorcycle samples are large because of the different postures, scales of motorcycles. Therefore, the large within-class gap and the large between-class gap lead to a great amount of false detection in the first stage. In the helmet detection stage, there are black helmets and black hair, hats and helmets, which are similar in colour and shape, often leading to false detection. In order to solve these difficult samples, we introduced the attention mechanism to optimize the feature extraction ability of the network, and effectively improve the detection accuracy.

In recent years, attention mechanisms have been applied to various tasks of computer vision. Non-local [45] captures long-range features by calculating dependencies between features in different locations. SENet [46] proposed channel attention, which explicitly modelled dependencies between channels. GCNet [47] optimizes Non-local, combines global attention with channel attention. SKNet [48] uses a channel attention mechanism to achieve a dynamic selection of field perception. Convolutional block attention module (CBAM) [49] combines channel and spatial attention mechanisms in sequence mode.

We experimented with several different attention mechanisms and chose to add triplet attention [50] to the last layer of the backbone network. Attention modules that fuse both spatial and channel-wise attention usually model spatial and channel-wise dependencies separately which leads to a lack of semantic interaction between different dimensions of features. Triplet attention extracts the semantic dependence between different dimensions, eliminates the indirect correspondence between channels and weights, and achieves the effect of improving accuracy with little computational overhead. Figure 4 shows the basic structure of triplet attention. Triplet attention uses three parallel branching structures, two of which extract the inter-dimensional dependencies between two spatial dimensions and channel dimension C, and the other extracts the spatial feature dependencies. In the first two branches, triplet attention rotates the original input tensor 90° counter-clockwise along the H-axis and W-axis respectively, and transforms the shape of the tensor from C × H × W to W × H × C and H × C × W. In the third branch, the tensor is entered in its original shape C × H × W. After that, the tensor of the first dimension is reduced to the second dimension through the Z-pool layer, and the average aggregation feature and the maximum aggregation feature are connected. The Z-pool is defined as:

$$Z-Pool\ (x) = \left[MaxPool_{od}\ (x)\ , AvgPool_{od}\ (x)\right]. \tag{1}$$

Then, the reduced tensor is passed through the standard convolution layer with a kernel size of K, batch normalization layer, and finally, the attention weight of the corresponding dimension generated by the sigmoid function is added into the rotated tensor. At the final output, the output of the first branch rotates 90 degrees clockwise along the H axis and the output of the second branch rotates 90 degrees clockwise along the W axis, ensuring the same shape as the input. Finally, the output of the three branches is aggregated equally as the output. The output tensor is defined as:

$$y = \frac{1}{3}\left(\overline{\hat{x}_1\sigma\left(\varphi_1\left(\widehat{x_1^*}\right)\right)} + \overline{\hat{x}_2\sigma\left(\varphi_2\left(\widehat{x_2^*}\right)\right)} + x\sigma\left(\varphi_3\left(\widehat{x_3^*}\right)\right)\right). \tag{2}$$

3.1.3 | Soft-NMS for final processing of the results

NMS is applied to most state-of-the-art detectors to obtain the final results because it significantly reduces the number of false positives. The flow chart of the NMS algorithm is shown in Figure 5a. First, all the detection boxes in the list are sorted according to their confidence scores. Second, the detection box B_M with the highest score is moved to the final detection list D, and the remaining detection boxes are assigned a unique identifier B_i. Third, any prediction box B_i whose overlap area with B_M is greater than a certain threshold N_t is removed. Repeat this process for the remaining boxes B_i until the initial list is empty. However, motorcycles block each other on crowded roads, and the NMS algorithm will lead to missed detection. This problem

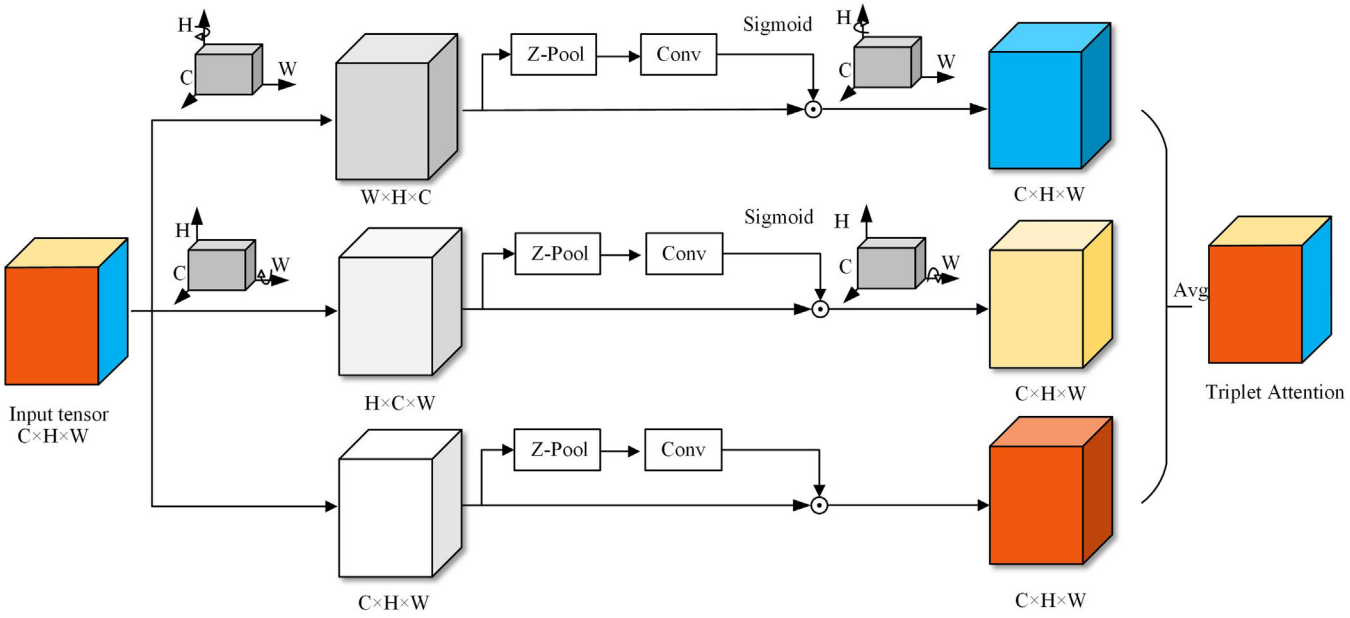

FIGURE 4 Triplet attention structure

is illustrated in Figure 5b. The blue and green boxes represent the detection results of different objects and correspond to different confidence scores. Suppose the blue box gets the highest score, followed by green and red. If the NMS algorithm is used, the green box will be removed because it has a large overlap with the blue box. Therefore, the motorcycle predicted by the green box will be ignored. So we introduced Soft-NMS instead of NMS to process the detection results. The NMS algorithm can be represented by rescoring function, as shown in the following formula (3):

$$S_i = \begin{cases} S_i, & IoU\,(B_M, B_i) < N_t \\ 0, & IoU\,(B_M, B_i) \geq N_t \end{cases} \tag{3}$$

where S_i is the score of the prediction box B_i,B_M is the prediction box with the highest score, and N_t is the overlap threshold. In NMS, a hard threshold is set to determine which boxes should be retained and which boxes should be deleted in B_M domain. If an object actually exists but has an overlap rate with B_M more than N_t, its detection will be ignored. The core idea of soft-NMS is to use a penalty function to attenuate the scores of prediction boxes that overlap with B_M rather than setting these scores to zero. The S_i of soft-NMS is shown in the following formula (4):

$$S_i = \begin{cases} S_i, & IoU\,(B_M, B_i) < N_t \\ S_i e^{-\frac{IoU(B_M,B_i)^2}{\sigma}}, & IoU\,(B_M, B_i) \geq N_t \end{cases} \tag{4}$$

where $e^{-\frac{IoU(B_M,B_i)^2}{\sigma}}$ is a Gaussian penalty function, and σ is a super parameter selected according to experience. It is obvious that the score of the prediction box with a large overlap with B_M will be greatly reduced, while the detection box far away from B_M will not be affected. If the score of the prediction box is still higher than the confidence threshold O_t after penalty. Then the prediction box will be retained rather than discarded. Using soft-NMS to deal with motorcycles in crowded scenes can greatly reduce the missed detection rate of motorcycles.

4 | MOTORCYCLE DETECTION

At present, YOLO series algorithms have been widely used in the field of intelligent transportation because of their high precision and high speed, such as license plate recognition [15]. The latest version of YOLOv5 has better performance than all previous versions. Therefore, we take YOLOv5 as the basic model of motorcycle detection, and by modifying the depth and width of the backbone network, modifying the output of the network, integrating triplet attention, improving NMS, using K-means++ to recalculate the anchor size, we call the model used in this stage YOLOv5-MD.

The dataset for this stage was obtained from the traffic surveillance video. In the traffic scene, there are cars, motorcycles, bicycles, tricycles and other types of vehicles. We find that bicycles, forward-looking tricycles and motorcycles are similar in their riding state. In [32] and [33], they only consider one motorcycle category, which will cause a lot of false detections. Therefore, in order to reduce the false detection rate, we detect three categories at this stage, that is, motorcycle, bicycle and tricycle. At the same time, we found that the number of bicycle and tricycle images is relatively small, so we use data enhancement methods, such as flipping, translating, blurring, etc., to expand a small number of categories.

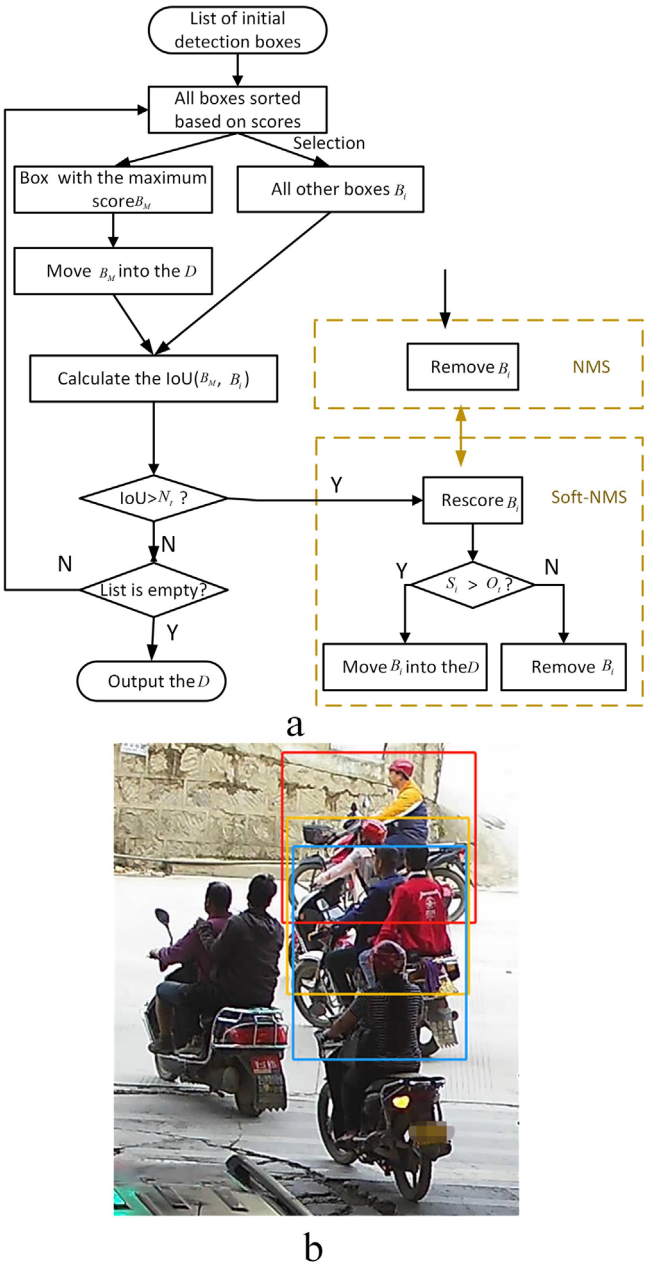

FIGURE 5 (a) The algorithm flow of NMS and Soft-NMS. (b) An illustration of the problem

The data of this stage is from the original image of traffic monitoring, which has a high resolution, and also has the following challenges:

1. The size of motorcycles at different distances from the camera varies greatly.
2. The images taken in different scenes have different viewing angles, such as profile, front, back and so on.
3. The illumination of images collected at different time intervals is different.
4. The challenges of different weather, such as rain and snow.
5. The challenges of crowded scenes, such as motorcycles blocking each other and other vehicles blocking each other.

In order to overcome the above challenges, first of all, we use a larger model, the depth_multiple and width_multiple parameters are set to 0.67 and 0.75 respectively, and triplet attention is added at the end of the backbone network to extract features of motorcycles to the greatest extent. Then, considering the real-time performance, we choose 640 × 640 network input size and adopt a multi-scale training strategy. Finally, the filter number of the last convolution layer is modified to match the number of categories. YOLOv5 uses the same output as YOLOv3, with three branches, and each branch matches three anchor boxes. Each anchor box uses four coordinates (x, y, w, h), confidence and C category probabilities. Therefore, the number of filters per branch is:

$$filters = (C + 5) \times 3. \tag{5}$$

The detection category in this stage is three, so the convolution kernel number of the final convolution layer is 24 ((3 + 5) × 3).

Since motorcycle detection is similar to other object detection tasks, we do not start training from scratch. We used the pre-trained model obtained from the COCO dataset to fine tune the YOLOv5-MD model. In the training process, when the overlap ratio IOU between the prediction box and ground truth box is more than 0.5, it is regarded as a positive sample, otherwise, it is a negative sample. In the test phase, we choose different confidence thresholds and IOU thresholds for soft-NMS to test, and finally select the threshold that takes into account the recall rate and accuracy rate as our best result.

4.1 | Helmet detection

The task of this stage is to detect whether the motorcyclist is wearing a helmet. The training data of this stage is the motorcycle image including the motorcyclist, which is cut from the original traffic monitoring image. The ratio of width to height of the image is about 1:2, the position of helmet or head is generally in the upper part of the motorcycle image, and the image size is smaller, which is less difficult than motorcycle detection. Therefore, in order to meet the real-time, we choose a smaller model, the depth_multiple is 0.33, the width_multiple is 0.5. The input size of the network is 320 × 320. There are two detection categories in this stage: helmet and no_helmet. Therefore, the convolution kernel number of the final convolution layer is 21 ((3 + 5) ×3). The dataset of this stage is quite different from the first stage, so we continue to use the K-means++ algorithm to re-cluster the size of the anchor. We call the model of this stage YOLOv5-HD. The main challenges at this stage are as follows:

1. The black helmet and the head with black hair may be mistakenly detected.
2. A head with a hat may be mistakenly detected as wearing a helmet.
3. When there are multiple riders on the motorcycle, the head of the passenger sitting behind may be blocked.

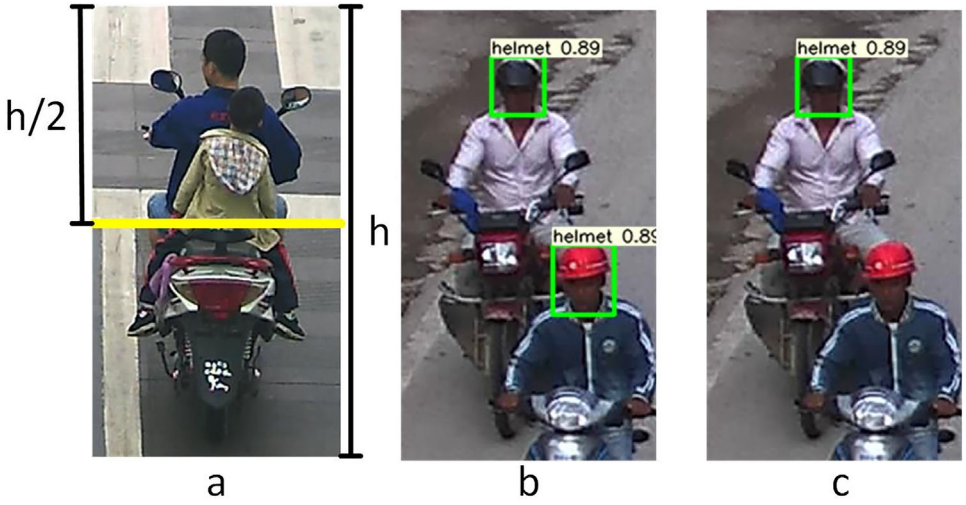

FIGURE 6 (a) Shows a motorcycle with children on it. (b) Shows the detection result without processing. **(c)** Shows the detection result after processing

4. For the detected motorcycle areas in the crowded scene, there may be other motorcyclists or pedestrians.

Aiming at the challenges of (1)–(3), we adopt the strategy of using data enhancement technology to expand such difficult samples, so as to reduce the false detection rate and missed detection rate. For the problem (4), as shown in Figure 6a, we find that the head of the motorcyclists are always in the top half of the image (1/2 of the image height), although there are children, so if it is detected that the head is in the bottom half of the image, it is most likely that the head belongs to other motorcycles or pedestrians, we will abandon it. As shown in Figure 6b, it is the motorcycle area detected in the crowded scene, and other motorcycles appear in the bottom half of the image. After our post-processing, we can effectively exclude other heads, as shown in Figure 6c.

5 | HFUT-MH DATASET

As far as we know, although many motorcycle helmet detection methods have been proposed, the datasets used are small and not comprehensive. In the currently published dataset, the UMD dataset [18] is only collected in one place, there is no diversity; OSF dataset [51] is obtained from 12 observation points of seven cities in Myanmar using cameras, which lack night images, and the resolution is 1920 × 1080, which is not obtained in traffic monitoring scenes. In view of the lack of available data for the actual traffic scene, we have built a large-scale dataset, named HFUT-MH. The dataset was obtained from several road monitoring systems in different cities in China, including two sub-datasets, HFUT-MH1 and HFUT-MH2, with a total of 53,126 images. Some sample images of HFUT-MH are shown in Figure 7. HFUT-MH1 is the original image obtained from the surveillance video, including 28,634 images, a total of three categories, the number of tags are 26,902 motorcycles, 13,912 bicycles and 11,840 tricycles, which are used for motorcycle detection. HFUT-MH2 is cut from the original image according to the location information of the motorcycle label in the first stage. It contains 24,492 pictures in two categories. The number of labels is 14,013 helmets and 14,583 heads respectively. It is used to further detect whether the motorcyclist is wearing a helmet when a motorcycle is detected. Our annotation tool is labelling, which is used to manually annotate the two sub-datasets to generate the annotation file containing the coordinates and category information of the object. The file format is XML.

In HFUT-MH1, in order to improve the diversity of data, we give up a lot of repeated images, which mainly come from the adjacent frames in the video. For non-riding vehicles, such as those parked on the roadside, we do not label them. For highly blurred objects far away from the surveillance camera, we do not label them. In HFUT-MH2, a lot of repeated images are also discarded, and the highly blurred images that are difficult to separate the helmet from the head are also discarded. The HFUT-MH dataset has the following properties:

1. The images are obtained from multiple traffic monitoring scenes, with different lighting conditions, such as day and night; different weather conditions, such as rainy and snowy days; different view angles, such as front, back and profile; different congestion levels.
2. The dataset consists of two subsets: HFUT-MH1 for motorcycle detection and HFUT-MH2 for helmet detection.
3. The dataset is the actual image obtained from traffic monitoring, and the trained model is more suitable for traffic scenes.

6 | EXPERIMENTAL RESULTS

In this section, we verify the effectiveness of the proposed method through experiments. All the training and testing tasks in this paper are carried out on a computer with an AMD

FIGURE 7 (a) Are some original pictures of traffic monitoring, which are used for motorcycle detection. The first line shows images with different lighting conditions, the second line shows images with different weather conditions, and the third line shows images with different congestion conditions. (b) are examples of motorcycle pictures used for helmet detection, which are cut out from the original images and contain different perspectives, different illumination, different occlusion and different pixels. The dataset is limited by privacy, and the face and license plate in the image are mosaic

Ryzen 7 3700 × 8-Core Processor 3.6 GHz CPU, 16 GB of RAM and an NVIDIA GeForce RTX 2080 Super. The selected algorithm is implemented in the framework of PyTorch. The network's initial learning rate is set to 0.01. The optimization algorithm adopts stochastic gradient descent (SGD), momentum parameter is 0.937, attenuation coefficient is 0.0005, and batch size is 16. The iteration number of model training is 300 epochs. Finally, we randomly divided HFUT-MH1 and HFUT-MH2 datasets into a training set and test set, with a ratio of 7:3.

The YOLOv5 algorithm introduces an anchor box, which makes the network only need to predict the offset between the ground truth box and anchor box, so as to improve the stability of the training. The anchor box represents the prior knowledge obtained from the statistics of training samples, so choosing a reasonable anchor box size can effectively improve the detection performance of the model for some unknown size and shape of objects. However, the original anchor box size of YOLOv5 is calculated by the K-means algorithm in the COCO dataset, and the HFUT-MH dataset proposed is quite

different from the COCO dataset. If we use the original anchor box directly for training, it will have a great impact on the training time and accuracy. Therefore, it is necessary to recalculate the size of the anchor box. Compared with the K-means algorithm, K-means++ reduces the influence of initial value selection on clustering results, and makes the loss function converge faster. The distance measure used in the clustering process is shown in formula (6):

$$D(box, centroid) = 1 - IoU(box, centroid) \tag{6}$$

where box represents the boundary box area, centroid represents the cluster centre, and the calculation formula of IOU is shown in (7):

$$IoU = \frac{A \cap B}{(A \cup B)}. \tag{7}$$

We use the same number of anchor boxes as YOLOv3. HFUT-MH1 dataset is re-clustered by K-means++, and the sizes of nine anchor boxes are: (15, 41), (23, 60), (30, 87), (41, 59), (41, 115), (64, 89), (93, 196), (107, 285), with an average intersection over union of 0.81. HFUT-MH2 dataset is re-clustered by K-means++, and the sizes of nine anchor frames are: (41, 44), (57, 39), (58, 48), (66, 45), (71, 50), (72, 40), (82, 52), (95, 55), with an average intersection over union of 0.91.

6.1 | Evaluation metrics

In order to quantitatively evaluate the detection effect of the model, we use the following criteria to detect the performance:

recall and precision

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

where TP (true positive) indicates the number of correctly detected objects, FP (false positive) indicates the number of wrongly detected objects, and FN (false negative) represents the number of missed objects, that is, the objects that should be detected are not detected.

F1-score

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{10}$$

It is the harmonic average of accuracy and recall, and is a comprehensive criterion to evaluate the detection performance.

AP

$$AP = \sum_{i=1}^{n} precision(k) \times \Delta recall(k). \tag{11}$$

where n is the total number of images in the dataset, precision (k) is the precision at a cutoff point of k images, and Δrecall (k) is the difference in recall between the cutoff point k-1 and the cutoff point k. Average the AP of each category again to get mAP. In this paper, we refer to the Pascal VOC evaluation criteria. When the IOU of the detection box and ground truth box is 0.5, we think it is a positive sample, otherwise it is a negative sample.

FPS is the frame rate per second, which represents the number of images that can be processed per second. It is usually used to evaluate the speed of object detection. But the performance of different hardware is different, so the FPS of different algorithms need to be evaluated on the same hardware.

6.2 | Motorcycle detection results

6.2.1 | Comparison with state-of-the-art methods

In order to verify the good performance of our proposed YOLOv5-MD algorithm, we compared it with other state-of-the-art object detection algorithms. In the evaluation experiments of other algorithms, we use the MMdetection framework, which has opened up most of the current object detection algorithms. And we all choose the model with the highest accuracy, for example, Resnext101 is mostly selected as the backbone network. In order to ensure fairness, we use a unified input size of 640 × 640, unified training super parameters and a unified training platform. We test different confidence thresholds and IOU thresholds to detect as many motorcycles as possible, while avoiding a high false detection rate. Finally, the confidence threshold is 0.45 and the IOU threshold is 0.5. As shown in Table 1, the accuracy of our method is higher than that of other state-of-the-art algorithms, with mAP reaching 98.5% and FPS reaching 126. The mAP of the two-stage method and most one-stage methods are lower than 98%, and only ATSS, FoveaBox, SSD, YOLOv3, v4 achieve real-time results. The speed of our method is only a little lower than that of YOLOv5s, but it still achieves the effect of real-time detection.

At this stage, we detected three categories, namely motorcycles, bicycles and tricycles. The detection performance of each category is shown in Table 2. It can be seen from the table that the average precision (AP) of motorcycle is 98.4%, and the F1 score is 94.0%, reaching a high detection accuracy. In order to more intuitively reflect the detection effect of three types of vehicles, we draw the P-R curve, as shown in Figure 8 We can see that when R is close to 1, P is also close to 1.

6.2.2 | Experimental results of integrating different attention mechanisms

Table 3 compares the performance of adding different attention mechanisms at the end of the backbone network. We take the

TABLE 1 Performance comparison with state-of-the-art methods in HFUT-MH1

Methods	Backbone	mAP (%)	FPS
Two-stage:			
Faster R-CNN [4]	Resnext101	95.6	11
Cascade R-CNN [52]	Resnext101	95.6	18
Libra R-CNN [53]	Resnext101	96.7	22
Grid R-CNN [54]	Resnext101	96.4	23
Mask R-CNN [5]	Resnext101	96.5	16
Dynamic R-CNN[55]	Resnet50	95.7	10
One-stage:			
FCOS [10]	Resnext101	97.0	24
FreeAnchor [56]	Resnext101	96.2	17
RepPoints [11]	Resnext101	96.2	16
PAA [57]	Resnext101	97.0	16
ATSS [9]	Resnet101	95.9	39
FoveaBox [58]	Resnet101	88.8	40
FSAF [59]	Resnext101	97.3	23
VFNet [60]	Resnext101	97.0	21
SSD512 [7]	Vgg16	95.7	32
RetinaNet [8]	Resnext101	96.2	22
YOLOv3 [34]	Darknet53	96.4	60
YOLOv4 [61]	CSPdarknet	97.0	71
YOLOv5s [12]	CSPdarknet	98.2	**133**
YOLOv5m [12]	CSPdarknet	98.2	110
YOLOv5l [12]	CSPdarknet	98.3	69
YOLOv5-MD (Ours)	CSPdarknet	**98.5**	126

TABLE 2 The detection results of motorcycle, bicycle and tricycle

Metrics	Motorcycle	Bicycle	Tricycle	Total
P	96.0	97.4	93.5	95.3
R	92.5	97.6	92.9	94.4
AP	98.4	99.3	97.9	98.5
F1-score	94.0	97.5	93.1	94.9

TABLE 3 Results of integrating different attention mechanisms

Method	AP (%)	FPS
Baselin6	98.23	**133**
Baseline +SE [46]	98.43	130
Baseline +GC [47]	98.45	123
Baseline +SK [48]	98.44	121
Baseline +CBAM[49]	98.46	131
Baseline +Triplet Attention [50]	**98.50**	126

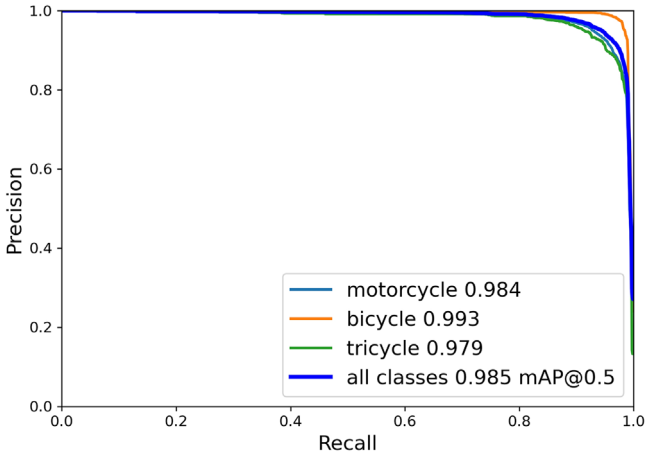

FIGURE 8 P-R curves of three categories of vehicles

TABLE 4 mAP value for soft-NMS under different σ values

σ	mAP (%)
baseline	98.41
0.01	98.36
0.0s	98.48
0.10	**98.50**
0.15	98.45
0.20	98.42

network without the attention mechanism of YOLOv5-MD as the baseline. From the data in the table, we can get the highest detection accuracy by adding triplet attention, with an AP of 98.5% and FPS of 126. Although the speed is reduced a little, the accuracy is improved by 0.27%.

6.2.3 | Detection results in crowded scenes

In the traffic scene, congestion often occurs, especially in high periods of commuting or in countries with a large number of motorcycles. When the road is congested, motorcycles are easy to be blocked. In order to improve the detection performance of motorcycles in crowded scenes, we introduced soft-NMS post-processing. We use the results obtained by the NMS algorithm as the baseline, and compare the detection results of soft-NMS under different super parametersσ, as shown in Table 4. We can see that when σ is 0.1, the mAP reaches the highest level, which is 98.5%.

Figure 9 shows the detection of motorcycles in nine different crowded scenarios. There are two kinds of occlusion in crowded scenes: motorcycles are occluded by other objects (such as cars), and motorcycles are occluded by each other. From Figure 9, we can see that when a motorcycle is partially blocked by other vehicles or by other motorcycles, it can still be detected

FIGURE 9 Detection results of nine different crowded scenes

correctly, which proves that our algorithm has good detection performance in crowded scenes.

6.3 | Helmet detection results

In order to verify the good performance of our proposed YOLOv5-HD algorithm, we compared it with other state-of-the-art object detection algorithms. As with the motorcycle detection stage, we also use the MMdetection framework to evaluate other algorithms. In order to ensure fairness, we use a unified input size 320 × 320, unified training super parameters and a unified training platform. By testing different confidence thresholds and IOU thresholds, in order to achieve the optimal detection accuracy, the final confidence threshold is 0.47 and the IOU threshold is 0.6. As shown in Table 6, the accuracy of our method is higher than that of other state-of-the-art algorithms, with mAP reaching 99.2% and FPS reaching 135. The speed of our method is only a little lower than that of YOLOv5s, but it still achieves the effect of real-time detection.

In this stage, we detected two categories, namely, helmet and *No*_helmet, As shown in Table 5, without considering the accuracy of one-stage detection, the AP of the helmet is 99.5%, and the AP of the No_helmet is 98.9%, both of them have high detection accuracy. In order to show the detection effect more intuitively, we drew the P-R curve, as shown in Figure 10.

TABLE 5 The detection results of helmet and No_helmet

Metrics	Helmet	No_helmet	total
P	97.7	98.2	98.0
R	99.0	97.1	97.2
AP	99.5	98.9	99.2
F1-score	94.0	97.5	97.6

Figure 11 shows the results of helmet detection. As shown in the first column of the figure, we can see that when a black helmet and black hair exist at the same time, our model can distinguish them correctly. As shown in the second column of the figure, the head of the passengers behind the motorcycle is almost covered by more than half, and our method can still successfully detect all the targets. However, if the classification algorithm is used, it is difficult to determine whether there are people without helmets on the motorcycles. In addition, there

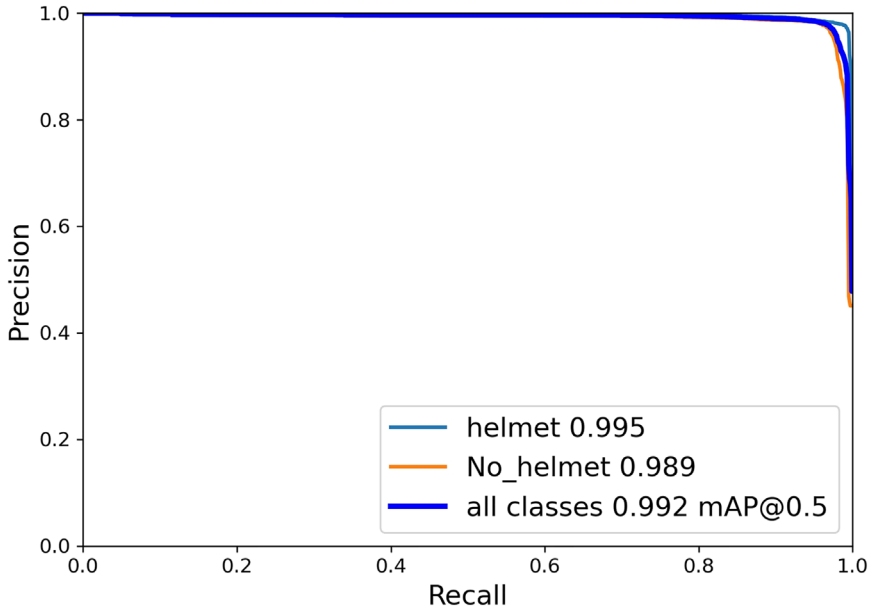

FIGURE 10 P-R curves of helmet and No_helmet

FIGURE 11 The results of helmet detection

may be other motorcyclists in the motorcycle image detected in the crowded scene. After our post-processing, we can successfully abandon other motorcyclists. For motorcycles from different perspectives, such as front and back, we can also achieve high detection performance.

Table 7 reports the detection accuracy and speed of each stage and the final end-to-end detection results. It can be seen from the table that the final detection performance of our proposed method: mAP is 97.7%, F1-score is 92.7%, FPS is 63, which achieves high accuracy and real-time effect.

TABLE 6 Performance comparison with state-of-the-art models in HFUT-MH2

Methods	Backbone	mAP(%)	FPS
Two-stage:			
Faster R-CNN [4]	Resnext101	98.5	14
Cascade R-CNN [52]	Resnext101	98.6	21
Libra R-CNN [53]	Resnext101	98.5	25
Grid R-CNN [54]	Resnext101	98.4	26
Mask R-CNN [5]	Resnext101	98.6	25
Dynamic R-CNN [55]	Resnet50	98.6	21
One-stage:			
FCOS [10]	Resnext101	98.1	28
FreeAnchor [56]	Resnext101	98.5	22
RepPoints [11]	Resnext101	98.5	18
PAA [57]	Resnext101	98.3	20
ATSS [9]	Resnet101	98.5	43
FoveaBox [58]	Resnet101	98.6	48
FSAF [59]	Resnext101	98.6	28
VFNet [60]	Resnext101	98.1	26
SSD512 [7]	Vgg16	98.3	36
RetinaNet [8]	Resnext101	98.4	27
YOLOv3 [34]	Darknet53	98.5	68
YOLOv4 [61]	CSPdarknet	98.9	76
YOLOv5s [12]	CSPdarknet	99.0	**141**
YOLOv5m [12]	CSPdarknet	99.1	109
YOLOv5l [12]	CSPdarknet	99.2	60
YOLOv5-MD (Ours)	CSPdarknet	**99.2**	135

TABLE 7 End to end motorcycle helmet detection results

Detection stage	mAP (%)	F1-score (%)	FPS
Motorcycle detection	98.5	94.9	126
Helmet detection	99.2	97.6	135
Total	**97.7**	**92.7**	**63**

7 | CONCLUSION

In this paper, we introduce a real-time end-to-end helmet detection of motorcyclists method based on YOLOv5 algorithm. This method can automatically detect the motorcycle in the video or image, and judge whether the rider on the motorcycle is wearing a helmet. Our method includes two stages of motorcycle detection and helmet detection, and for each stage, we train a model, which are YOLOv5-MD and YOLOv5-HD, to achieve a real-time effect while ensuring high accuracy. In addition, our model size is still very advantageous, the first stage model is only 20.6 MB, and the second stage model is only 14.8 MB. In order to verify the effectiveness of our method, we also propose a new motorcycle helmet dataset, HFUT-MH, which is obtained from multiple traffic scenes in China with a variety of complex weather conditions, different occlusion conditions and different light conditions. In our dataset, the final end-to-end motorcycle helmet detection mAP reached 97.7%, F1-score reached 92.7, detection speed reached 63 FPS, achieved high accuracy and real-time effect. In addition, our method can determine whether the motorcycle is overloaded by calculating the number of helmets and *No*_helmets. In the future, we may add a tracking algorithm to it, and detect the same object only once to avoid repeated detection, which is very necessary for actual traffic video surveillance.

ACKNOWLEDGEMENTS

This work is partly supported by the grants of the National Natural Science Foundation of China, Nos. 62076086, 61673157, 61972129, 61972127, and partly supported by the grants of the Key Research and Development Program in Anhui Province 202004d07020008 and 201904d07020010.

ORCID

Ye Yu https://orcid.org/0000-0001-5628-6237

REFERENCES

1. Organisation mondiale de la santé: Global Status Report on Road Safety 2018, WHO (2018)
2. WHO Sri Lanka: World Health Organization. https://www.who.int/srilanka. accessed 2021.
3. Wang, Q.: Weakly supervised adversarial domain adaptation for semantic segmentation in urban scenes. IEEE Trans. Image Process. 28(9), 4376–4386 (2019)
4. Ren, S., at el.: Faster R-CNN: towards real-time object detection with region proposal networks. arXiv:1506.01497 [cs], (2016)
5. He, K., et al.: Mask R-CNN. arXiv:1703.06870 [cs], (2018)
6. Redmon, J., et al.: You only look once: unified, real-time object detection. arXiv:1506.02640 [cs], (2016)
7. Liu, W., et al.: SSD: single shot multiBox detector. arXiv:1512.02325 [cs], 9905, 21–37 (2016)
8. Lin, T.-Y., et al.: Focal loss for dense object detection. arXiv:1708.02002 (2018)
9. Zhang, S.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020
10. Tian, Z., et al.: FCOS: Fully convolutional one-stage object detection. arXiv:1904.01355 [cs], (2019)
11. Yang, Z., et al.: RepPoints: Point set representation for object detection. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 27 Oct.–2 Nov. 2019
12. ultralytics/yolov5, https://github.com/ultralytics/yolov5, accessed 2021
13. Wang, Q., et al.: Embedding structured contour and location prior in siamesed fully convolutional networks for road detection. IEEE Trans. Intell. Transport. Syst. 19(1), 230–241 (2018)
14. Guo, S., et al.: Attention based spatial-temporal graph convolutional networks for traffic flow forecasting'AAAI. 33, 922–929 (2019)
15. Laroca, R., et al.: A robust real-time automatic license plate recognition based on the YOLO detector. In: International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018
16. Sun, D., et al.: Visual cognition inspired vehicle re-identification via correlative sparse ranking with multi-view deep features. In: Ren, J., et al. (Eds.): Advances in Brain Inspired Cognitive Systems Springer International Publishing, Xi'an, China, (2018)
17. Bodla, N., et al.: Soft-NMS – Improving object detection with one line of code. arXiv:1704.04503 [cs], (2017)

18. Espinosa, J.E.: Detection of motorcycles in urban traffic using video analysis: A review. IEEE Trans. Intell. Transport. Syst. 1–16 (2020)
19. Silva, R., et al.: Automatic detection of motorcyclists without helmet. In: 2013 XXXIX Latin American Computing Conference, Caracas, Venezuela, 7–11 Oct. 2013
20. Dahiya, K., et al.: Automatic detection of bike-riders without helmet using surveillance videos in real-time. In: 2016 International Joint Conference on Neural Networks (IJCNN)' 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016
21. Padmini, V.L., et al.: Real time automatic detection of motorcyclists with and without a safety helmet. In: 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 10–12 Sept. 2020
22. Swapna, M.: A hybrid approach for helmet detection for riders safety using Image processing. Mach. Learn. Artif. Intell. 182(37), 50–55 (2019)
23. Marayatr, T., Kumhom, P.: Motorcyclist's helmet wearing detection using image processing. Adv. Mat. Res. 931–932, 588–592 (2014)
24. Silva, R.R.V.e.: Detection of helmets on motorcyclists. Multimedia Tools Appl. 77(5), 5659–5683 (2018)
25. Talaulikar, A.S.: An enhanced approach for detecting helmet on motorcyclists using image processing and machine learning techniques. In: Mandal, J.K., (Eds.): Advanced Computing and Communication Technologies, Springer, Singapore (2019)
26. Contractorr, D.: Cascade classifier based helmet detection using open CV in image processing. (2016)
27. Zheng, A., et al.: Local-to-global background modeling for moving object detection from non-static cameras. Multimedia Tools Appl. 76(8), 11003–11019 (2017)
28. Li, C., et al.: Moving object detection via robust background modeling with recurring patterns voting. Multimedia Tools Appl. 77, (11), 13557–13570 (2018)
29. Shine, L.C.V.J.: Automated detection of helmet on motorcyclists from traffic surveillance videos: A comparative analysis using hand-crafted features and CNN. Multimedia Tools Appl 79(19–20), 14179–14199 (2020)
30. Vishnu, C., et al.: Detection of motorcyclists without helmet in videos using convolutional neural network. In: 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017
31. Yogameena, B.: Deep learning-based helmet wear analysis of a motorcycle rider for intelligent surveillance system. IET Intell. Transp. Syst. 13(7), 1190–1198 (2019)
32. Allamki, L., et al.: Helmet detection using machine learning and automatic license plate recognition. International Research Journal of Engineering and Technology 06(12), 5 (2019)
33. Khan, F.A.: Helmet and number plate detection of motorcyclists using deep learning and advanced machine vision techniques. In: 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 15–17 July 2020
34. Redmon, J., Farhadi, A.: YOLOv3: an Incremental Improvement. arXiv:1804.02767 [cs], (2018)
35. Saumya, A., et al.: Machine learning based surveillance system for detection of bike riders without helmet and triple rides. In: 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 10–12 Sept. 2020
36. Rohith, C.A., et al.: An efficient helmet detection for MVD using deep learning. In: 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 23–25 April 2019
37. Chairat, A., et al.: Low cost, high performance automatic motorcycle helmet violation detection. In: 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Snowmass, CO, USA, 1–5 March 2020
38. Santhosh, A.: Real-time helmet detection of motorcyclists without helmet using convolutional neural network'IJRASET. International Journal for Research in Applied Science and Engineering Technology 8(7), 1112–1116 (2020)
39. Dasgupta, M.: Automated helmet detection for multiple motorcycle riders using CNN. In: 2019 IEEE Conference on Information and Communication Technology, Allahabad, India, 6–8 Dec. 2019
40. Lin, H., et al.: Helmet use detection of tracked motorcycles using CNN-based multi-task learning. IEEE Access 8, 162073–162084 (2020)
41. Siebert, F.W., Lin, H.: Detecting motorcycle helmet use with deep learning. Accident. Anal. Prev. 134, 105319 (2020)
42. Boonsirisumpun, N.: Automatic detector for bikers with no helmet using deep learning. In: 2018 22nd International Computer Science and Engineering Conference (ICSEC), Chiang Mai, Thailand, 21–24 Nov. 2018
43. Wang, C.-Y., et al.: CSPNet: A new backbone that can enhance learning capability of CNN. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020
44. Wang, K., et al.: PANet: Few-shot image semantic segmentation with prototype alignment'no date. 10
45. Wang, X., et al.: Non-local neural networks. arXiv:1711.07971 [cs], (2018)
46. Hu, J.: Squeeze-and-excitation networks'no date. 10
47. Cao, Y., et al.: GCNet: Non-local networks meet squeeze-excitation networks and beyond. arXiv:1904.11492 [cs], (2019)
48. Li, X., et al.: Selective kernel networks. arXiv:1903.06586 [cs], (2019)
49. Woo, S., et al.: 'CBAM: Convolutional Block Attention Module. in Ferrari, V., (Eds.): Computer Vision – ECCV. Springer International Publishing, Munich, Germany, (2018)
50. Misra, D., et al.: Rotate to attend: Convolutional triplet attention module. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 Jan. 2021
51. Cai, Z., Vasconcelos, N.: Cascade R-CNN: delving into high quality object detection. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018
52. Pang, J., et al.: Libra R-CNN: towards balanced learning for object detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019
53. Lu, X., et al.: Grid r-cnn. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019
54. Zhang, H., et al.: Dynamic R-CNN: towards high quality Object detection via dynamic training. arXiv:2004.06002 [cs] (2020)
55. Zhang, X., et al.: FreeAnchor: learning to match anchors for visual object detection'no date. 9
56. Kim, K., Lee, H.S.: Probabilistic anchor assignment with IoU poredictionfor object detection. arXiv:2007.08103 [cs] (2020)
57. Kong, T., et al.: FoveaBox: beyound anchor-based object detection. IEEE Trans. Image Process. 29, 7389–7398 (2020)
58. Zhu, C.: Feature selective anchor-free module for single-shot object detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019
59. Zhang, H., et al.: VarifocalNet: an IoU-aware dense object detector. arXiv:2008.13367 [cs] (2020)
60. Bochkovskiy, A., et al.: YOLOv4: optimal speed and accuracy of object detection. arXiv:2004.10934 [cs, eess], (2020)

How to cite this article: Jia, W., et al.: Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector. IET Image Process. 15, 3623–3637 (2021). https://doi.org/10.1049/ipr2.12295