Impolite Pedestrian Detection by Using Enhanced YOLOv3-Tiny

Yanming Wang[1, 2, 3], Kebin Jia[1, 2, 3] and Pengyu Liu[1, 2, 3, *]

Abstract: In recent years, the problem of "Impolite Pedestrian" in front of the zebra crossing has aroused widespread concern from all walks of life. The traffic sector's governance measures have become more serious. The traditional way of governance is on-site law enforcement, which requires a lot of manpower and material resources and is low efficiency. An enhanced YOLOv3-tiny model is proposed for pedestrians and vehicle detection in traffic monitoring. By modifying the backbone network structure of YOLOv3-tiny model, introducing deep detachable convolution operation, and designing the basic residual block unit of the network, the feature extraction ability of the backbone network is enhanced. The improved model is trained on the VOC2007+VOC2012 training set, and the trained model is tested for performance on the test data set. The experimental results show that: the mean Average Precision (mAP) increased from 0.672 to 0.732, increasing the measurement accuracy by 9%. The Intersection over Union (IoU) increased from 0.783 to 0.855, increasing the coverage accuracy by 7.2%. The enhanced YOLOv3-tiny model has higher measurement accuracy than the original model. Applying this model to the 1080P traffic video on the NVIDIA RTX 2080, the detection speed is 150 FPS, which can fully achieve real-time detection. Through the analysis of pedestrians and vehicle coordinates, it is judged whether or not illegal acts occur. For illegal vehicles, save three pictures as the basis for law enforcement, which forms an important supplement to off-site law enforcement.

Keywords: Impolite pedestrian, YOLOv3-tiny, deep detachable convolution, residual block, off-site law enforcement.

1 Introduction

In front of the crosswalk, vehicles slow down and stop to let pedestrians go first. It's the basic traffic rules. However, the current situation is that most drivers have not yet developed this habit, especially at intersections without traffic lights, this behavior is more prominent. For this illegal behavior, the current detection methods are: a) on-site law enforcement; b) setting manual signal lights; c) setting speed reduction belts, etc. However, the methods either require a lot of labor costs, and lack of strong evidence of law

[1] Faculty of Information Technology, Beijing University of Technology, Beijing, 100124, China.

[2] Beijing Laboratory of Advanced Information Networks, Beijing, 100124, China.

[3] Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing, 100124, China.

[*] Corresponding Author: Pengyu Liu. Email: liupengyu@bjut.edu.cn.

Received: 13 February 2020; Accepted: 05 April 2020.

JAI. doi:10.32604/jai.2020.010137

enforcement, making these methods have little effect in practical applications [Al-masni, Al-antari, Park et al. (2018); Ahmetovic, Bernareggi, Gerino et al. (2014)]. In order to solve the above problems, this paper proposes an enhanced yolov3-tiny detection algorithm. It can comprehensively monitor the indecent behavior of motor vehicles and become an important supplement to the existing off-site law enforcement system.

The You Only Look Once (YOLO) method unifies target classification and positioning into a regression problem. The core idea of this method is to use the whole image as the input of the network, and directly return to the position of the bounding box and its associated category at the output layer [Joseph, Santosh, Ross et al. (2016); Joseph and Ali (2017); Corovic, Ilic and Duric (2018)]. Compared with the Faster R-CNN and SSD, the detection speed is greatly improved. Its latest version, YOLOv3, not only maintains the original detection speed, but also greatly increases the detection accuracy, making it the preferred target detection algorithm [Joseph and Ali (2018); Chen, He, Shi et al. (2019)].

YOLOv3-tiny is a simplified version of YOLOv3, which has the advantages of small amount of calculation, high real-time performance, and easy integration in embedded devices [Tian, Yang, Wang et al. (2019)]. The disadvantage is that the detection accuracy is low. The network structure adopts the traditional convolution + pooling form and the backbone network is shallow [Liu, Hou, Zhang et al. (2019)]. The higher level semantic features cannot be extracted. In this paper, the backbone network structure of YOLOv3-tiny model is modified, the deep detachable convolution operation is introduced, and the basic residual block unit of the network is designed to enhance the feature extraction ability of the backbone network [Zhang, Zhang, Yang et al. (2016)].

For the detection of illegal behavior, the first detection should be the zebra crossing area. According to the actual situation, the camera position is fixed, and the zebra crossing area is also fixed. By planning the zebra crossing area in advance, the coordinates of the zebra crossing can be determined more quickly and more stably. Vehicle and pedestrian detection are performed by using the YOLOv3-tiny algorithm to obtain the coordinates of the corresponding target, and then the coordinates are analyzed to determine whether illegal behavior occurs [Luo, Wang, Cai et al. (2019)].

The rest of the paper is organized as follows. The second section introduces zebra crossings, pedestrians, and vehicle detection. The third section describes how to detect illegal vehicles according to traffic rules. The fourth section introduces the relevant experiments and discusses the experimental results. Finally, summarize and look forward to this article.

2 Object detection

2.1 Zebra crossing detection

There are many detection methods for zebra crossing recognition based on image processing, such as bipolar number method, evanescent point method, and frequency domain feature method [Ahmetovic, Bernareggi, Gerino et al. (2014)]. These methods have high linearity requirements for zebra crossings and are susceptible to light, which will not correctly identify zebra crossings at night. According to the actual application, we can draw the position in the image where the zebra crossing is located in advance, and get the corresponding outline and position information. As shown in Fig. 1. By adopting this method, the influence of pedestrians, light and other factors on zebra crossing recognition can be avoided, which lays a foundation for subsequent detection.

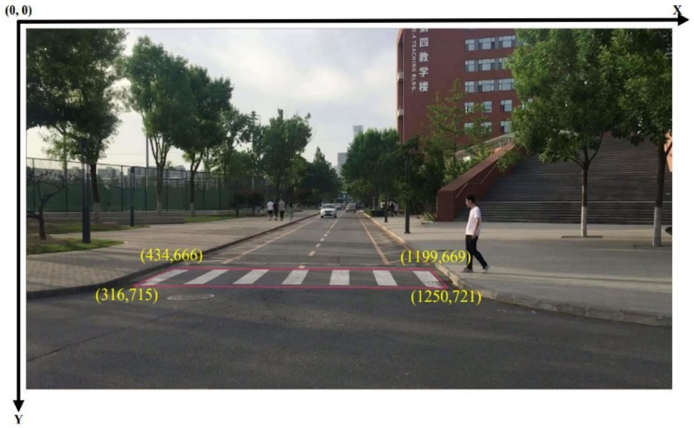

Figure 1: Identify the zebra crossing region

2.2 YOLOv3-tiny

The network divides each image into S × S grids. If the center of the target ground truth falls in a grid, then the grid is responsible for detecting the target. As shown in Fig. 2, an image is divided into 7×7 grids, and the (4, 4) grid is responsible for predicting this person [Ahmetovic, Bernareggi, Gerino et al. (2014)]. Each grid predicts B bounding boxes and their confidence scores, as well as C class conditional probabilities. When multiple bounding boxes detect the same target, YOLO uses the non-maximum suppression (NMS) method to select the best bounding box.

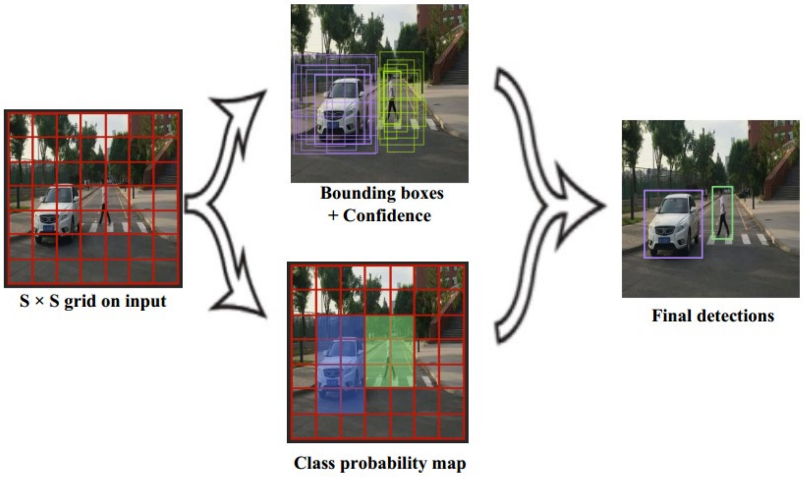

Figure 2: YOLO Detection

YOLO has two kinds of network structures. If high precision is required, YOLOv3 with complex network structure can be used. If real-time performance is required, YOLOv3-tiny can be used. The difference between YOLOv3 and YOLOv3-tiny is mainly in the following three aspects. First, the YOLOv3 backbone network uses a 53 layers convolutional network to extract features, and the YOLOv3-tiny backbone network uses a 7 layers convolutional layer + pooling layer. The second is that the YOLOv3 feature extraction network uses a lot of 1×1, 3×3 convolutional layers, and this part is greatly reduced in YOLOv3-tiny; The third is at the final YOLO output layer, YOLOv3 has a larger scale. According to the actual needs, the detected content is limited to cars and pedestrians and has high real-time requirements [Joseph and Ali (2018)]. In this paper, the backbone network structure of YOLOv3-tiny model is modified, the deep detachable convolution operation is introduced, and the basic residual block unit of the network is designed to enhance the feature extraction ability of the backbone network, so as to achieve the coexistence of real-time detection and high accuracy [Liu, Hou, Zhang et al. (2019)].

Residual network is a neural network model proposed by He et al. in 2016 [He, Zhang, Ren et al. (2016)]. It solves the problem that the network cannot converge due to the disappearance of gradient in the deep network. The network model is widely used in the network design unit because of its superior performance, and its basic residual structure block is shown in Fig. 3.

In the standard convolution, the feature graph will convolution with each filter, while in the depth detachable convolution, each feature graph will convolution with only one filter, which greatly reduces the amount of computation. The network performance can be greatly improved by multiplex image features. In this paper, the basic residual block and depth detachable convolution are combined to extract the features of the backbone network, and the basic structure unit is shown in Fig. 4.

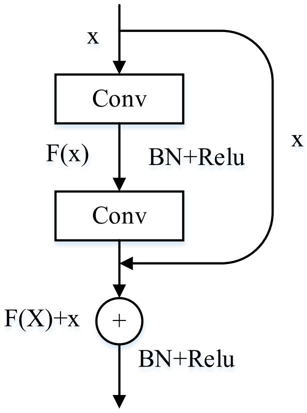

Figure 3: Residual Block

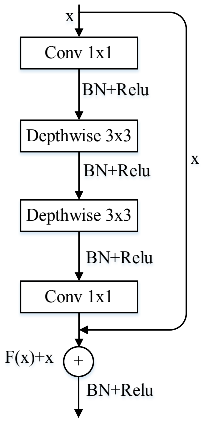

Figure 4: New Residual Block

In order to better process the high-resolution image, the input image is first adjusted to a size of 416×416 pixels, then the image is reduced to the original 1/32 by 5 down sampling. The YOLOv3-tiny model predicts the bounding box on two different scales of 13×13 and 26×26. It also classifies the target category and returns the coordinates of the bounding box and the confidence. The comparison of the backbone network before and after modification is shown in Tab. 1. The test results are shown in Fig. 5.

3 Methodologies

3.1 Definition of illegal behavior

In any case, when the motor vehicle is driving to the zebra crossing, it should be slowed down. If there are pedestrians on the zebra crossing, the vehicle must stop. Combined with relevant regulations, the typical scene of motor vehicle polite pedestrians is shown in Fig. 6. Motor vehicles in the front lane of pedestrians must stop. The lane that the pedestrian has passed, the motor vehicle can pass [Gabriel, Schleiss, Schramm et al. (2018)].

Table 1: Backbone Net Parameters Comparison

YOLOv3-tiny				Enhanced YOLOv3-tiny			
Type	Filters	Size	Output	Type	Filters	Stride	Output
Conv	16	3×3 / 1	416×416×16	Conv	16	1	416×416×16
Max		2×2 / 2	208×208×16	Block	16	2	208×208×16
Conv	32	3×3 / 1	208×208×32	Block	32	2	104×104×32
Max		2×2 / 2	104×104×32	Block	64	2	52×52×64
Conv	64	3×3 / 1	104×104×64	Block	128	2	26×26×128
Max		2×2 / 2	52×52×64	Block	256	2	13×13×256
Conv	128	3×3 / 1	52×52×128	Block	512	1	13×13×512
Max		2×2 / 2	26×26×128	Block	1024	1	13×13×1024
Conv	256	3×3 / 1	26×26×256				
Max		2×2 / 2	13×13×256				
Conv	512	3×3 / 1	13×13×512				
Max		2×2 / 1	13×13×512				
Conv	1024	3×3 / 1	13×13×1024				

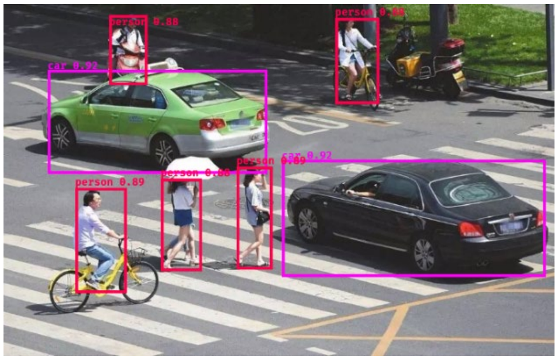

Figure 5: Detection result

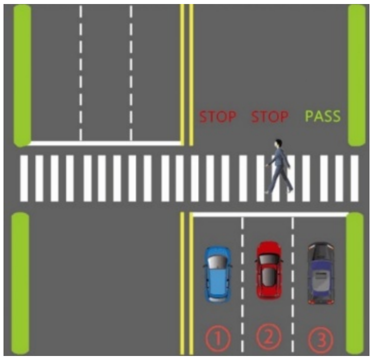

Figure 6: Polite pedestrian rule

3.2 Pedestrian analysis

When loading the first frame, each pedestrian in the image is labeled, the coordinates are recorded, and the running direction is determined. When the pedestrian first appears, if the position coordinates are to the left of the zebra crossing, the direction of motion is determined to be from the left to the right through the zebra crossing; if the position is to the right of the zebra crossing, the direction of motion is determined to be from the right to the left through the zebra crossing. If the pedestrian is already on the zebra crossing when starting the test, the initial coordinates are recorded first, and then the coordinates of the last few frames are compared with the initial coordinates to determine the direction of motion.

When the next frame is read, the coordinates are updated according to the number to obtain its real-time position information. When a new person appears in the image, the calculation process is the same as the first frame.

When all pedestrians are detected, personnel are required to filter to find people who are only walking on the zebra crossing. According to the pedestrian coordinates and the zebra crossing coordinates, it is judged whether the pedestrian is on the zebra crossing, if in the zebra crossing area, the relevant information of the person is retained, and if not, the person is removed from the list. The test results are shown in Fig. 7 and Fig. 8.

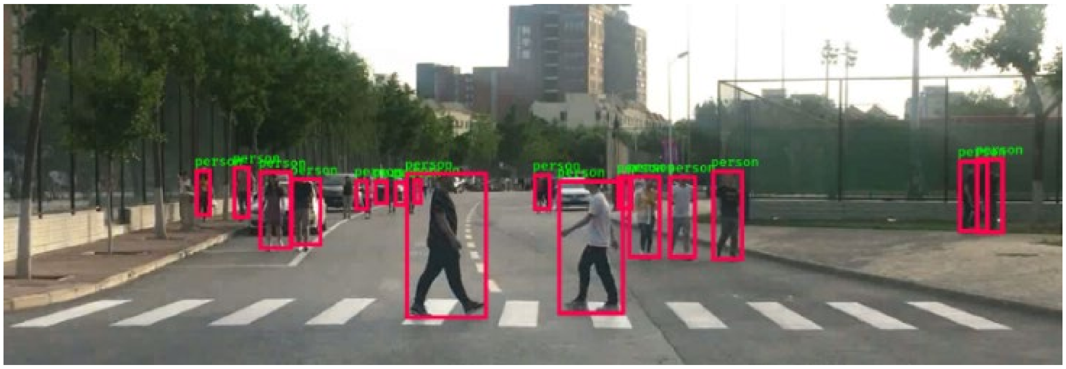

Figure 7: Unfiltered person test results

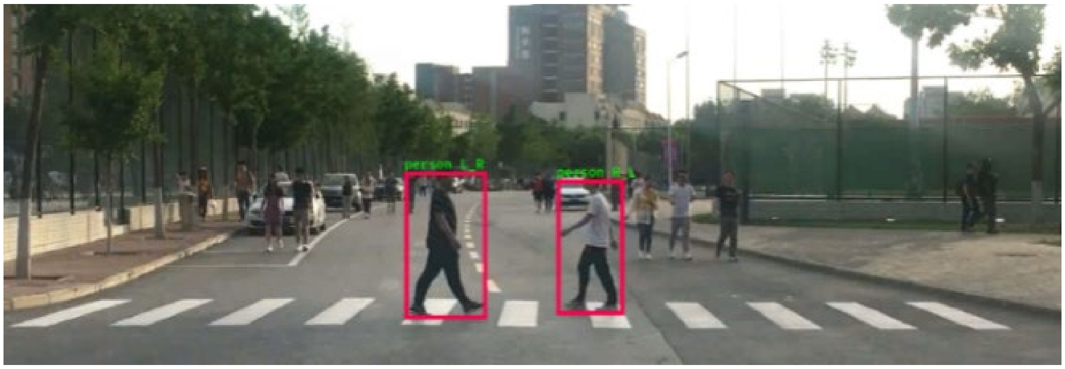

Figure 8: Filtered person test results (L_R: From left to right R_L: From right to left)

According to the traffic rules, the car is able to pass through the lane where the pedestrian has walked, so it is necessary to further divide the zebra crossing area to determine the area where the car can travel. As shown in Fig. 9 and Fig. 10.

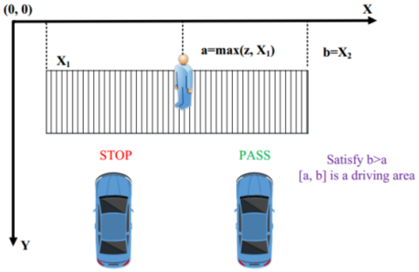

Figure 9: Satisfactory condition

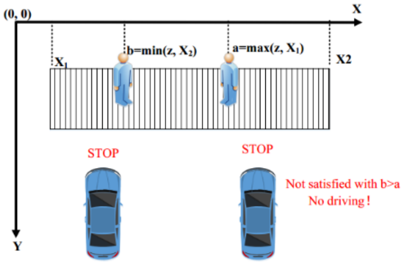

Figure 10: Unsatisfactory condition

We need to set two variables a, b and initialize them to the X-axis coordinates of the leftmost $x1$ and the rightmost $x2$ of the zebra crossing, and the pedestrian's X-axis coordinates are z. When a pedestrian moves from right to left, a is equal to the larger of z and $x1$. When a pedestrian moves from left to right, b is equal to the smaller of z and $x2$. When $b > a$, the $[a, b]$ area is the car passable area, otherwise, the car is forbidden to pass.

3.3 Vehicle analysis

After detecting the vehicle by YOLOv3-tiny, record its coordinates and running direction. Then, the coordinates are analyzed to filter out the vehicles near the zebra crossing. As shown in Fig. 11 and Fig. 12. Based on the width of the zebra crossing, if the vehicle coordinates are within this area, the vehicle information is retained. If not, the vehicle is removed from the list.

For the stored vehicle information, the driving area determination is performed. According to the analysis of Fig. 9 and Fig. 10, when the driving area is in $[a, b]$, no illegal behavior occurs, otherwise, illegality may occur. As shown in Fig. 10, when $X_1 > b$ or $X_2 < a$, there is illegal behavior. Keep information about this vehicle. Otherwise, there is legal behavior. Remove this vehicle from the list. Finally, the vehicles in the list may be illegal vehicles.

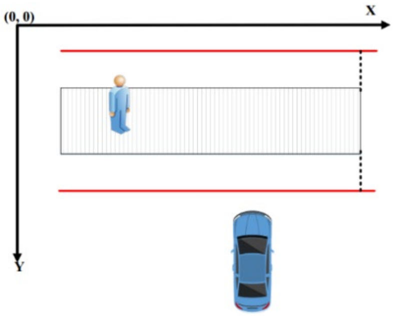

Figure 11: Filter vehicle

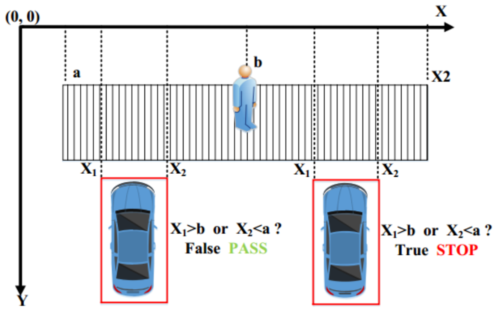

Figure 12: Judgment of illegal acts

For illegal vehicles, the system automatically records three photos as evidence of law enforcement. The test results are shown in Fig. 13~Fig. 15. (The label means category, direction. The purple rectangle is the drivable area.)

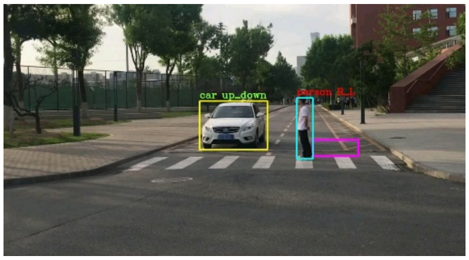

Figure 13: Outside the zebra crossing

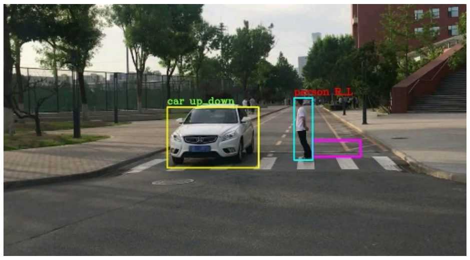

Figure 14: Passing the zebra crossing

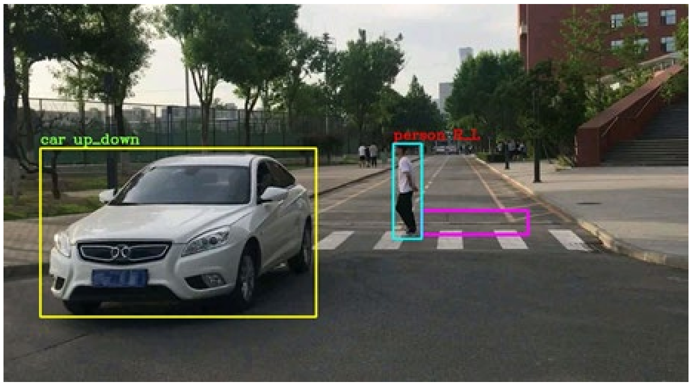

Figure 15: Go through the zebra crossing

4 Experimental results

The enhanced YOLOv3-tiny detection model used in this study was modified using the Darknet framework. The detection models were trained and tested on an NVIDIA RTX 2080. The network initialization parameters are shown in Tab. 2

Table 2: Initialization Parameters of Enhanced Yolov3-Tiny

Size of input images	Batch	Momentum	Initial learning rate	Decay	Training step
416×416	8	0.9	0.01	0.0005	60000

In order to improve the detection accuracy of the model and to adapt the input required for the Darknet framework, the input images were adjusted to 416×416 pixels. Taking into account the memory constraints of the GPU, the batch size was set to 8 in this paper. 60,000 training steps were used in order to better analyze the training process. Parameters such as momentum, initial learning rate, weight decay regularization, and other parameters referred to the original parameters in the YOLOv3-tiny model. The model was trained after defining the training parameters. The learning rate decreased to 0.001 after 30, 000 steps and to 0.0001 after 40, 000 steps.

In this paper, a series of experiments with the trained enhanced YOLOv3-tiny model were conducted with the test images to verify the performance of the algorithm. The related indicators for evaluating the effectiveness of the neural network models are as follows:

4.1 Loss function

Loss function is an important factor for evaluating the performance of a model. The loss function in YOLOv3-tiny is defined as follows:

$$Loss = Error_{coord} + Error_{iou} + Error_{cls} \tag{1}$$

The coordinate prediction error $Error_{coord}$ is defined as follows:

$$\begin{aligned} Error_{coord} &= \lambda_{coord} \sum_{i=1}^{3} \sum_{i=1}^{B} 1_{ij}^{obj} \left[\left(x_i - \hat{x}_i\right)^2 + \left(y_i - \hat{y}_i\right)^2 \right] \\ &+ \lambda_{coord} \sum_{i=1}^{S^2} \sum_{j=1}^{B} 1_{ij}^{obj} \left[\left(w_i - \widehat{W}_1\right)^2 + \left(h_i - \widehat{h}_i\right)^2 \right] \end{aligned} \tag{2}$$

where λ_{coord} is the weight of the coordinate error, $S^2$is the number of grids in the input image, and B is the number of bounding boxes generated by each grid. Referring to the original parameters in the YOLOv3-tiny model, $\lambda_{coord} = 5$, $S = 7$, and $B = 6$ were selected in this study. $1_{ij}^{obj} = 1$ denotes that the object falls into the jth bounding box in grid i, otherwise $1_{ij}^{obj} = 0$. $(\widehat{x_\iota}, \widehat{y_\iota}, \widehat{w_\iota}, \widehat{h_\iota})$ are values of the center coordinate, height, and width of the predicted bounding box. (x_i, y_i, w_i, h_i) are true values.

The IOU error $Error_{iou}$ is defined as follows:

$$Erroriou = \sum_{i=1}^{S^2} \sum_{j=1}^{B} 1_{ij}^{obj} \left(C_i - \hat{C}_i\right)^2 + \lambda_{noobj} \sum_{i=1}^{S^2} \sum_{j=1}^{B} 1_{ij}^{obj} \left(C_i - \hat{C}_i\right)^2 \tag{3}$$

where c refers to the class to which the detected target belongs. $p_i(c)$ refers to the true probability that the object belonging to class c is in grid i. $\widehat{p_\iota}(c)$ is the predicted value. The $Error_{cls}$ for grid i is the sum of classification errors for all the objects in the grid.

4.2 IOU

IOU is a standard for defining the detection accuracy of target objects. IOU evaluates the performance of the model by calculating the overlap ratio between the predicted bounding box and the true bounding box as follows:

$$Iou = \frac{S_{overlap}}{S_{union}} \tag{4}$$

where $S_{overlap}$ is the area of intersection of the predicted bounding box and the true bounding box. S_{union} is the area of the union of the two bounding boxes [Tian, Yang, Wang et al. (2017)].

In order to verify the performance of the proposed model, the VOC2012, VOC2007 training set training network, VOC2007 test set to test the network. The training set contains a total of 7,800 pictures of people and cars, and more training samples are generated by adjusting saturation, brightness, and hue. The enhanced YOLOv3-tiny model was compared with the original YOLOv3-tiny model by 2000 images to test the model effect.

Fig. 16 shows the original YOLOv3-tiny, introduced residual block in the training process and the new residual block loss proposed in this paper.

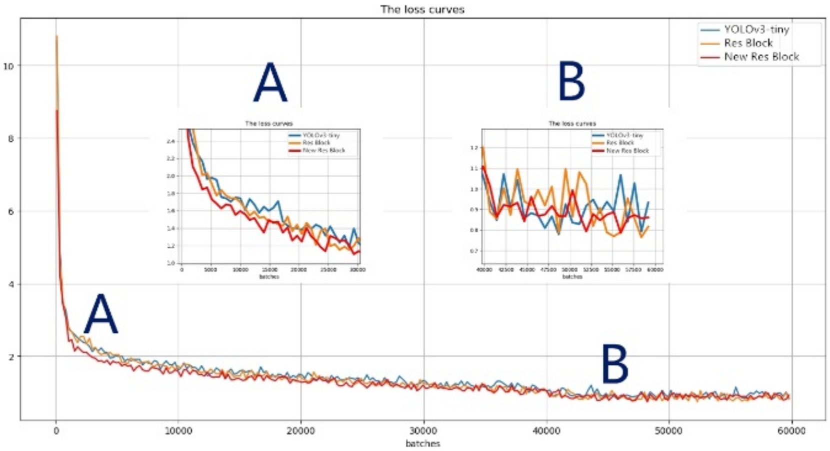

Figure 16: Loss curves of the three YOLO models

The P-R curves for several models during testing are shown in Fig. 17. The mean Average Precision (mAP), IOU, and frames per second (FPS) of the models are shown in Tab. 3.

Based on the above results, it can be seen that the residual block proposed in this paper converges faster than other models in the training process. The final losses of the three models were similar, about 0.943. In terms of detection performance, the enhanced YOLOv3-tiny model is better than the original tiny model. The mAP of enhanced YOLOv3-tiny model is 0.732 and the IOU value is 0.855, which are higher than other models. Although the enhanced YOLOv3-tiny model is 50 frames slower than the original

YOLOv3-tiny model, it still has good real-time detection ability. These experiments show that the performance of the proposed model is improved.

Table 3: mAP IOU and PFS For Several Models

Models	YOLOv3-tiny	Residual Block	New Residual Block
mAP	0.672	0.719	0.732
IOU	0.783	0.820	0.855
FPS	200	170	150

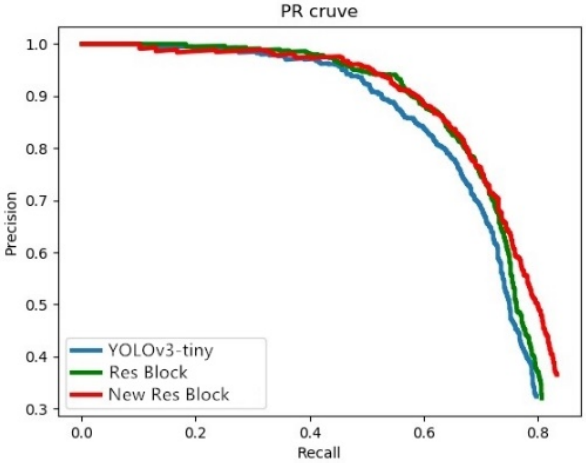

Figure 17: P-R curves for the detection models

5 Conclusions

This study introduces a neural network model for pedestrian and vehicle detection by detecting impolite pedestrian vehicles. Based on the original YOLOv3-tiny model, the performance of the network is improved by changing feature extraction and feature propagation. Compared with the original YOLOv3-tiny, the improved model has better performance. The model is used to detect illegal behavior, which not only improves the detection accuracy, but also ensures the real-time performance.

In the future work, traffic signal detection and license plate recognition functions can be introduced to better assist off-site law enforcement and improve the needs of actual scene detection. In addition, a deeper network structure can be introduced, and more training data can be used for model optimization to further improve detection accuracy.

Funding Statement: This paper is supported by the following funds: National Key R&D Program of China (2018YFF01010100), National natural science foundation of China (61672064), Beijing natural science foundation project (4172001) and Advanced information network Beijing laboratory (PXM2019_014204_500029).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Ahmetovic, D.; Bernareggi, C.; Gerino, A.; Mascetti, S. (2014): ZebraRecognizer: Efficient and precise localization of pedestrian crossings. *22th International Conference on Pattern Recognition*.

Al-masni, M. A.; Al-antari, M. A.; Park, J. M.; Gi, G.; Kim, T. Y. et al. (2018): Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system. *Computer Methods and Programs in Biomedicine,* vol. 157, no. 17, pp. 85-94.

Corovic, A.; Ilic, V.; Duric, S. (2018): The real-time detection of traffic participants using YOLO algorithm. *26th Telecommunications Forum*.

Chen, H. P.; He, Z. T.; Shi, B. W.; Zhong, T. (2019): Research on recognition method of electrical components based on YOLO V3. *IEEE Access*, no. 7, pp. 157818-157829.

Gabriel, E.; Schleiss, M.; Schramm, H.; Meyer, C. (2018): Analysis of the discriminative generalized Hough transform as a proposal generator for a deep network in automatic pedestrian and car detection. *Journal of Electronic Imaging*, vol. 27, no. 5, pp. 25-34.

He, K.; Zhang, X.; Ren, S.; Sun, J. (2016): Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 2016, pp. 770-778.

Joseph, R.; Santosh, D.; Ross, G.; Ali, F. (2016): You only look once: Unified, real-time object detection. *29th IEEE Conference on Computer Vision and Pattern Recognition*.

Joseph, R.; Ali, F. (2017): YOLO9000: Better, faster, stronger. *30th IEEE Conference on Computer Vision and Pattern Recognition*.

Joseph, R.; Ali, F. (2018): YOLOv3: An incremental improvement. https://arxiv.org/abs/1804.02767.

Liu, J.; Hou, S.; Zhang, K. (2019): Real-time vehicle detection and tracking based on enhanced Tiny YOLOV3 algorithm. *Transactions of the Chinese Society of Agricultural Engineering*, vol. 8, no.35, pp. 118-125.

Luo, M. H.; Wang, K.; Cai, Z. P.; Liu, A. F.; Li, Y. Y. et al. (2019): Using imbalanced triangle synthetic data for machine learning anomaly detection. *Computers, Materials & Continua*, vol. 58, no. 1, pp. 15-26.

Tian, Y. N.; Yang, Z. D.; Wang, H. (2017): Apple detection during different growth stages in orchards using the improved YOLO-V3 model. *Computers and Electronics in Agriculture*, vol. 157, no. 6, pp. 417-426.

Zhang, J.; Zhang, T.; Yang, Z. L.; Zhu, X. S.; Yang, B. X. (2016): Vehicle model recognition method based on deep convolutional neural network. *Transducer and Microsystem Technologies*, vol. 35, no. 11, pp. 19-22.