Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 7978644, 13 pages
https://doi.org/10.1155/2021/7978644

Research Article

SPDNet: A Real-Time Passenger Detection Method Based on Attention Mechanism in Subway Station Scenes

Jun Yang, Ying Zheng, KunPing Yan, HongJiang Liu, Kun Jin, WenLin Fan, Xiao Han, and YaWen Zhang

China University of Mining and Technology-Beijing, No. 11, Xueyuan Road, Haidian District, Beijing, China

Correspondence should be addressed to Ying Zheng; 516039700@qq.com

Received 23 September 2021; Accepted 29 October 2021; Published 7 December 2021

Academic Editor: Li Zhu

In order to implement real-time detection of passengers in subway stations, this paper proposes the SPDNet based on YOLOv4. Aiming at the low detection accuracy of passengers in the subway station due to uneven light conditions, we introduce the attention mechanism CBAM to recalibrate the extracted features and improve the robustness of the network. For the crowded areas in the subway station, we use the K-means++ algorithm to generate anchors that are more consistent with the passenger aspect ratio based on the dataset KITTI, which mitigates the missing caused by the incorrect suppression of true positive boxes by the Nonmaximum Suppression algorithm. We train and test our SPDNet on the KITTI dataset and prove the superiority of our method. Then, we carry out transfer learning based on the subway surveillance video dataset collected by ourselves to make it conform to the distorted passenger targets under the angle of the surveillance camera. Finally, we apply our network in a Beijing subway station and achieve satisfactory results.

1. Introduction

As an important method to relieve the pressure of urban traffic at present, the intelligent construction of subway has attracted more and more attention in the industry. Real-time detection of passengers based on a large number of video surveillance equipment in subway stations plays an important role in the construction of "Smart Subway." For urban rail transit dispatching, real-time detection of passengers in the station can be used to grasp the density of passenger flow in the station in time, which can also support the prediction of passenger flow [1]. Therefore, trains can be better deployed and the pressure of passenger transportation during peak hours can be alleviated. For the security in subway stations, accurate and rapid detection of passengers can effectively monitor the gathering situation of the crowds and then facilitate security dispatch to prevent possible safety accidents such as stampede. At present, the passenger detection method for the scene in the subway station is still relatively blank, and we urgently need a passenger detection method to solve such problems.

The environment in the subway station is complex; the uneven light conditions in the scene bring interference to the task of passenger detection. At the same time, the angle of the monitoring equipment in the subway station distorts the passenger target, which is different from the frontal human target in the conventional datasets. And the diversity of the placement angles produces passengers of different scales, among which passengers of smaller scales are easy to be ignored. Additionally, the real-time passenger detection task also makes higher requirements on the speed indicator of the detection method.

Passenger detection can also be regarded as pedestrian detection in the scene of the subway station. The traditional pedestrian detection method uses manual design methods to extract human features and then conducts a classifier to detect pedestrians, which has the disadvantages of large computation, poor robustness, and low precision. With the development of deep learning model, researchers introduce convolutional neural network (CNN) into the objection detection task, thus driving the development of pedestrian detection methods. Among them, the pedestrian detection

method based on pure CNN has become the mainstream method due to its characteristics of being easy to train, high accuracy, and good generalization ability. In these methods, YOLOv4 [2], as the master of all kinds of tricks in recent years, has a high speed and high precision, achieving a good balance between speed and precision, and possesses a high engineering application value. However, in the subway scene, due to the interference of light conditions, monitoring equipment deployment angle, and other factors, the original network performs a poor accuracy in the passenger detection task.

In response to the above problems, on the basis of YOLOv4 [2], we introduced the Convolutional Block Attention Module [3] to recalibrate the detection features of the channel dimension and the space dimension. This allows the network to focus on enhancing passenger characteristics and suppressing interference characteristics. In this way, the influence of light conditions on the detection task of passengers in subway stations is reduced. And the capturing ability of network on small-scale passengers caused by the deployment angle of monitoring equipment is also improved, Thus, we further improve the detection accuracy of the network. In addition, K-means++ [4] algorithm is conducted to design the anchors of YOLOv4 [2], making it more suitable for the size of passenger targets, so as to alleviate the omission caused by the postdetection nonmaximum suppression (NMS) algorithm falsely suppressing the true positive box in the case of dense crowds. To distinguish it from the existing approach, we name it Subway Passengers Detection Net (SPDNet).

In the KITTI dataset [5], we train and test our network and also compare it with multiple object detection methods. The experimental results show that our method has a better detection effect than other methods without affecting the real-time performance. We train the network based on a dataset of passengers in subway stations that we collect and label. Through migration learning technology, from the perspective of monitoring equipment, the network is adapted to distorted passenger detection. Finally, we apply our network to a subway station in Beijing and satisfactory results are obtained.

2. Related Work

2.1. Objection Detection. Objection detection [6] is one of the basic problems of current computer vision, and also an important basis for solving other computer vision tasks such as instance segmentation and target tracking. The objection detection task has experienced two phases: the method based on manual feature construction and the method based on the deep learning model.

Before 2014, objection detection algorithms were mostly based on manual feature construction, known as traditional objection detection methods. Viola and Jones proposed the VJ detector [7], which adopts the idea of sliding window and is combined with three strategies of accelerating computation and achieves real-time detection of human faces for the first time. Felzenszwalb et al. proposed DPM (Deformable Part-Based Model) [8], and then, Girshick et al. made various improvements to it [9–12]. DPM adopts the idea of divide and conquer, reaching the peak of traditional objection detection methods.

However, after 2012, the performance of manual design features tends to be saturated, and the model performance of traditional methods is difficult to make breakthrough progress.

Jeffrey Hinton designs AlexNet [13], the first convolutional neural network used in the image classification task, which achieves far more than the traditional image classification methods. Convolutional neural network automatically learns the features of images at various levels through convolution and pooling operations and simulates the process of human visual cortex recognition of images.

Compared with traditional manual design features, it is more flexible and accurate. Therefore, the rapid development of convolutional neural network soon replaces the traditional methods and has been widely used to solve various problems in computer vision.

Girshick et al. first apply convolutional neural network to the objection detection task and propose the RCNN [14] method, which pioneered the two-stage method in objection detection methods. In order to improve the detection accuracy and speed of RCNN, many extended versions are proposed, such as Fast RCNN [15], Faster RCNN [16, 17], and Mask RCNN [18]. Since the two-stage detection algorithm adopts a multistage detection scheme, it achieves a high detection accuracy. But the redundant calculation generated during the generation of candidate regions makes its interference speed slow. So it is not suitable for real-time objection detection tasks.

As an alternative method, Redmon et al. propose the first one-stage method, You Only Look Once (YOLO) [19]. It applies neural network in the entire input image and directly regresses all kinds of information of the bounding box in the output layer, greatly improving the inference speed.

Liu et al. propose the second one-stage detector Single Shot MultiBox Detector (SSD) [20] in the era of deep learning based on the anchor idea of Faster RCNN. Later, people develop YOLOv2 [21], YOLOv3 [22], YOLOv4 [2], DSSD [23], FSSD [24], and other networks on the basis of these two networks. The one-stage detector model has a high interference speed to meet the requirements of the real-time objection detection task. However, because it abandons the process of generating proposal regions, its positioning accuracy is poor, especially for the detection effect of small-scale targets deteriorated seriously, so the pure one-stage detector is also difficult to adapt to the complex environment in the scene of the subway station.

2.2. Pedestrian Detection. Passenger detection is a special case of pedestrian detection, and its scene is limited to the field of public transportation such as subway. Pedestrian detection [25], as one of the important tasks of computer vision, is also a subtask of objection detection task. Compared with objection detection, pedestrian detection is more focused on the application level, with stricter requirements on operational efficiency, false detection rate, and recall rate. In addition, extreme environmental conditions should be

taken into account for the task of pedestrian detection, so that it can be more robust in practical applications. At present, pedestrian detection task has important application value in artificial intelligence system, vehicle auxiliary driving system, intelligent robot, intelligent video surveillance, and other fields.

In the early stage, people use the manual design method to extract human features and conduct a classifier to detect pedestrians. Hand-crafted features play an important role in early pedestrian detection methods. The Histogram of Oriented Gradient (HOG) algorithm [26] proposed by Dalal and Triggs follows the ideas of multiscale pyramid and sliding window and is a key milestone in pedestrian detection methods, on which many subsequent methods are extended. At present, the traditional manual feature is mature and the performance tends to be saturated, but the traditional objection detection algorithm still has the disadvantages of large amount of computation and poor robustness, which is difficult to meet the requirements of application.

With the increasing maturity of deep learning technology, people begin to explore the application of deep learning method to pedestrian detection and have achieved good results. Pedestrian detection methods based on deep learning can be roughly divided into two categories: hybrid method and pure CNN method. The hybrid method uses both manual design method and deep learning convolutional network for feature extraction. For example, [27] uses the combination of manually constructed pedestrian local body features and deep learning features to solve pedestrian occlusion problem. Other hybrid methods use hand-designed features to make proposal generation and then use deep learning convolutional network to classify proposals. For example, [28] uses predesigned pedestrian features of different scales to improve the recognition effect of pedestrians of different scales.

Compared with the hybrid method, the pure CNN method avoids the manual design of features, achieves the proposal generation and classification at the same time, and is simpler and more effective. In the pure CNN method, the pedestrian detection method based on the two-stage detector model has high accuracy, but it is slow and requires high hardware, which is difficult to meet the needs of deployment in application scenarios. On the other hand, the pedestrian detection method based on a single-stage detector model achieves real-time performance, but its robustness is poor, and its accuracy is seriously reduced in complex scenes, which means it is difficult to directly use CNN for pedestrian detection to achieve the expected effect. Therefore, according to the different detection tasks, targeted improvement of CNN is still needed. Lu et al. [29] train the double-output branch deep network to detect the whole body and visible part of the pedestrian, respectively, and then combine the results of the two parts to improve the pedestrian detection rate in the case of occlusion [30] that introduces a supplementary subnetwork to generate a high-resolution feature map for small-scale pedestrian detection, thus improving the detection accuracy of a small-scale pedestrian. However, the existing methods are generally complex, and although they have achieved good results on relevant datasets, they are hard to meet the requirements [31–33] of application in a subway station. Therefore, a simple, efficient, and fast method is still needed to solve the problem of passenger detection in subway stations.

2.3. The Development of YOLO. As the origin of a one-stage detector in the field of object detection and the most popular detector model at present, the design idea of the YOLO network originated from extending the basic CNN idea from the classification task to the detection task. Taking single objection detection as an example, the previously proposed RCNN regards the detection task as an ergodic classification task, which traverses all positions of the image and classifies each window by the sliding window method. The problem of this method is obvious. If the traversal is not complete, its accuracy is low, and the more accurate the traversal is, the longer time it takes. On the other hand, this kind of method is essentially training the classifier that takes the content of the sliding window as the input. Due to the idea of traversal, the problem of unbalanced category is very serious too. Therefore, the author of YOLO changes his focus. Instead of using the classifier to output the one-hot vector representing the category, the author directly regresses the position of the bounding box, which greatly improved the detection speed. On the basis of this idea, YOLOv1 [19] network is developed.

YOLOv1 is improved on the basis of GoogleNet [34] and forms a basic network for feature extraction. By dividing the input images into multiple grids, the bounding box is predicted by regression for each grid, to achieve the effect of multicategory and multitarget recognition. In the meantime, two bounding boxes, one large and one small, are predicted in each grid, which fully considers the prediction problem of multiscale objects. YOLOv1 is extremely fast and can satisfy the demand of real-time performance. Simultaneously, due to its comprehensive consideration of the global information contained in the picture, its generalization performance is outstanding. However, YOLOv1 also has many disadvantages, such as poor positioning accuracy and low recall rate, especially for small target objects; the positioning accuracy drops very fast. In addition, the recognition effect is unsatisfactory in the case of dense objects, and the recognition rate of abnormal objects with aspect ratio is low.

YOLOv2 [21] abandons the method of directly predicting bounding box in YOLOv1 and uses a small offset to predict, which achieves the effect of stable neural network training. Concurrently, in view of the missed detection of many small objects in YOLOv1, YOLOv2 refines the mesh division. What is more, YOLOv2 draws on the design idea of SSD [20] and connects high-resolution feature images with low-resolution feature images to realize the multiscale detection method for the whole process.

Although YOLOv2 has increased the number of predicted boxes, the problem of small objection detection is still not well solved. So YOLOv3 [22] tries to design three detection heads to predict three different levels of targets, large, medium, and small. Combined with the introduction of Feature Pyramid Network (FPN) [35] structure to extract

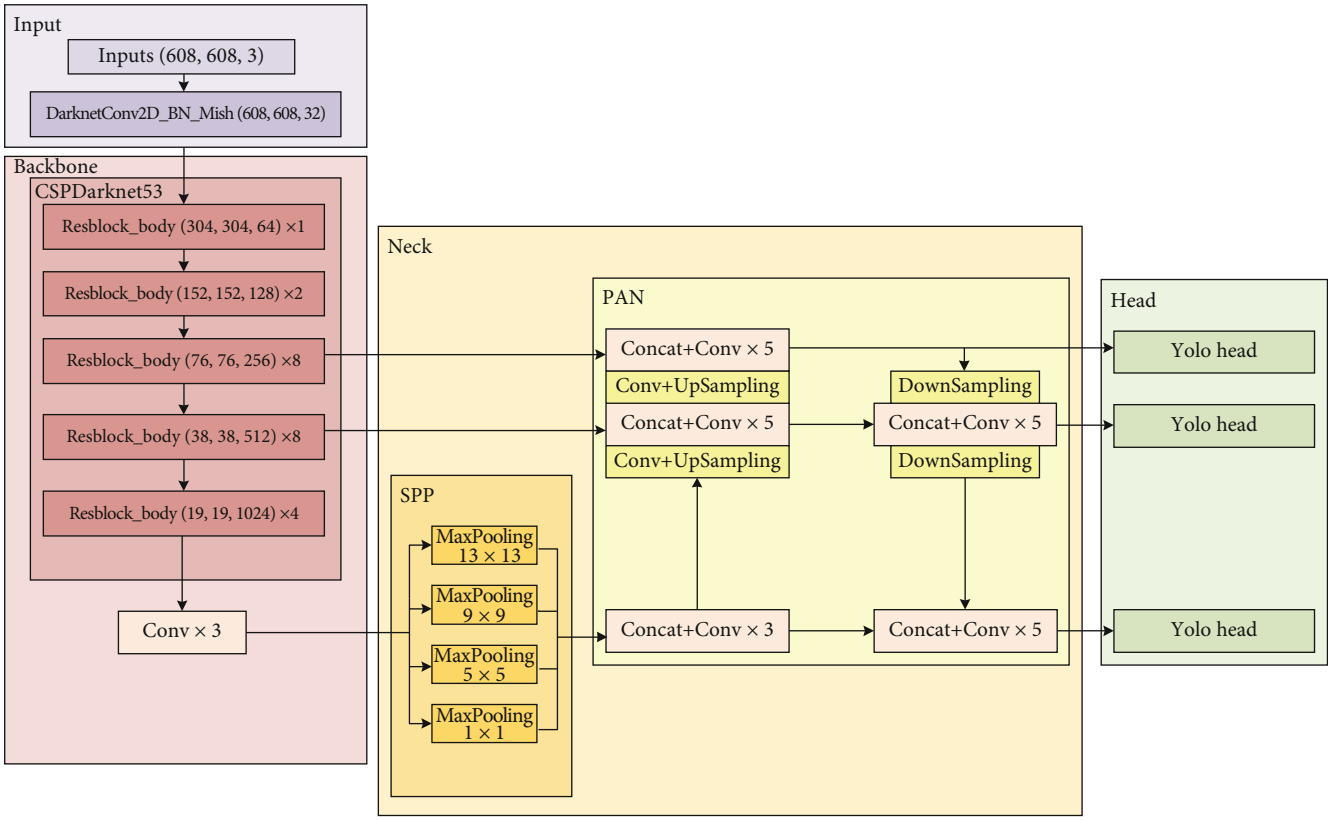


Figure 1: YOLOv4 network structure.

features from different scales and make independent prediction, a more accurate and efficient multiscale detection method is formed. Synchronously, YOLOv3 introduces residual module [36] into the backbone network to further deepen the number of network layers, thus extracting deeper features of images and achieving better learning effects.

YOLOv4 [2] is the epitome of many tricks in recent years and is widely used in the industry due to its ease of use. At present, the YOLO series network has been updated to the fifth generation. Compared with the fourth generation network, YOLOv5 has reduced the number of network model parameters and improved the detection speed through a slight drop in accuracy and has made an initial appearance in the field of target detection.

3. Proposed Method

3.1. YOLOv4 Structure. Although the YOLO series network has been updated to the fifth generation, the YOLOv5 model has not been officially released, and its model is not yet stable. It is still being continuously updated on the open source community GitHub. Considering that the YOLOv4 network already meets the real-time requirements of subway passenger detection tasks and has slightly higher accuracy and robustness compared to the current YOLOv5 network, the more stable model YOLOv4 is selected as the baseline for this network.

YOLOv4 is a powerful and efficient detection model designed on the basis of YOLOv3. It retains the overall structure of YOLOv3 and improves each substructure by using many state-of-the-art tricks in the field of object detection in recent years, involving input layer, backbone network, neck network, loss function, and other aspects. While retaining its high interference speed, the network accuracy is further improved to achieve a good balance between speed and accuracy. At the same time, YOLOv4 reduces the training difficulty of the network, making it possible to train on a single GPU such as 1080Ti or 2080Ti. These factors also make YOLOv4 of high application value in the engineering field, suitable for deployment in the actual production and life.

YOLOv4 can be divided into four parts, namely, input, backbone, neck, and head, as shown in Figure 1.

At the input layer, YOLOv4 innovates the data-enhanced Mosaic algorithm, which uses random scaling, clipping, and arranging of four images to join together, greatly enriching the detection dataset. In particular, random scaling adds many small targets and enhances the robustness of the network. In addition, the author can directly calculate the data of four images when using the Mosaic method for data enhancement, which reduces the size of the minibatch and simplifies the network training.

YOLOv4 uses CSPDarknet53 as the backbone network. The network is based on the deep learning framework Darknet designed by YOLO author Joseph Redmon. By introducing the residual module, the training difficulty caused by the depth of the network is solved, so as to increase the depth of the backbone network and form a

stronger structure containing 53 convolution layers, finally reaching a better effect of extracting the deeper features of the object. At the same time, Cross Stage Partial (CSP) thought [37] is used to optimize each residual module, which further enhances the learning ability of the convolutional neural network, ensures the accuracy of the network, and reduces the computing bottleneck and memory cost of the network while realizing the lightweight network structure.

In the neck network, inspired by SPPNet [38], YOLOv4 designs the SPP (Spatial Pyramid Pooling) module, which realizes the feature map level fusion of local features and global features, increases the network's region of interest, and enriches the expression ability of the final feature map. Meanwhile, YOLOv4 adopts PAN [39] structure, which adds or connects adjacent feature graphs from top to bottom according to the order of elements, fully integrates semantic features and positioning features between each feature pyramid layer, and realizes parameter aggregation of different head layers from different backbone layers. It enriches the information of afferent neural network head.

The head network of YOLOv4 is mainly responsible for detecting tasks. The YOLOv4 algorithm does not need to generate region of interest (ROI) in advance but trains the network directly in a regression way. In the head network, YOLOv4 uses three detectors, which generate 13×13, 26×26, and 52×52 grids on the input image, respectively, and predict three bounding boxes for each grid. In the meantime, three groups of predefined bounding box sizes are preset on three scales, which is called anchor, and subsequent positioning prediction is carried out based on the preset anchor of nine sizes, which achieves the effect of detecting targets at three scales severally. After that, the detection head predicts a vector P for each bounding box, which is composed as follows:

$$P = \left(t_x + t_y + t_w + t_h\right) + P_0 + (C_1 + C_2 + \cdots + C_n). \quad (1)$$

The first four elements in the vector P are related to the coordinates of the bounding box and determine the position and size of the final predicted box. The correspondences are

$$\begin{aligned} b_x &= \text{Sigmoid}(t_x) + d_x, \\ b_y &= \text{Sigmoid}\left(t_y\right) + d_y, \\ b_w &= p_w \times e^{t_w}, \\ b_h &= p_h \times e^{t_h}, \end{aligned} \quad (2)$$

where d_x and d_y represent the offset of the grid to the upper left corner of the picture to which the bounding box belongs and p_w and p_h are the length and width of the predefined anchors. b_x and b_y represent the position of the final predicted result's bounding box relative to the upper left corner of the picture.

b_w and b_h represent the length and width of the final predicted bounding box.

The fifth element P_0 in the vector P stands for confidence which is calculated by

$$P_0 = \text{Prob}(\text{object}) \times \text{IoU}_{\text{object}}^{\text{truth}}. \quad (3)$$

Prob(object) represents the probability that the object is in the predicted box, and $\text{IoU}_{\text{object}}^{\text{truth}}$ represents the intersection and combination ratio between the predicted box and the ground-truth box. The probability of the object being in the predicted box is 1 when the highest score is given to the predicted box using logistic regression, and 0 otherwise. The remaining n values in the vector P represent the fraction that predicts the object to belong to one of the n classes.

The core idea of the YOLO series algorithm is to directly regression predict a certain number of bounding boxes for each grid in the input image, which greatly improves the detection speed of the network, but also inevitably leads to the problem of missed detection of small target objects. That is to say, the YOLOv4 network has limited processing capacity for small-scale passengers due to the placement angle of the monitoring equipment in the scene of the subway station, which needs further improvement. Simultaneously, the robustness of the YOLOv4 network needs to be further enhanced in the face of the interference of various light conditions in the subway station.

3.2. SPDNet Algorithm Principle. Based on the YOLOv4 network, this paper specifically integrates the attention mechanism module CBAM to calibrate the weighted features on the channel dimension and focus the features on the spatial dimension. The missing object location information on the channel dimension is supplemented through the spatial relationship of features, thus improving the capture effect of small-scale passengers, in order to solve the problem of small-scale passenger detection caused by the deployment angle of monitoring equipment in the scene of subway station. Meanwhile, the attention mechanism module CBAM enhances the characteristics of passengers and suppresses the interference characteristics in the environment, thus reducing the impact of different light conditions on the passenger detection task in the subway scene, and further improving the passenger detection accuracy of the network. Additionally, K-means++ algorithm clustering is adopted in this paper to design the size of anchor according to the passenger scale features, so that it can better fit the passenger target and alleviate the suppression of the true positive boxes caused by the NMS algorithm in the case of dense crowds. The improved network structure SPDNet is shown in Figure 2.

3.3. Anchor Optimization. YOLOv4 network presets three types of bounding boxes, a total of nine, called anchors, whose sizes are (12,16), (19,36), (40,28), (36,75), (76,55), (72,146), (142,110), (192,243), and (459,401). It is used to predict the boundary box with the detection heads of 76×76, 38×38, and 19×19, respectively.

These anchors are obtained by clustering on the Pascal VOC dataset of regular size targets, covering targets with

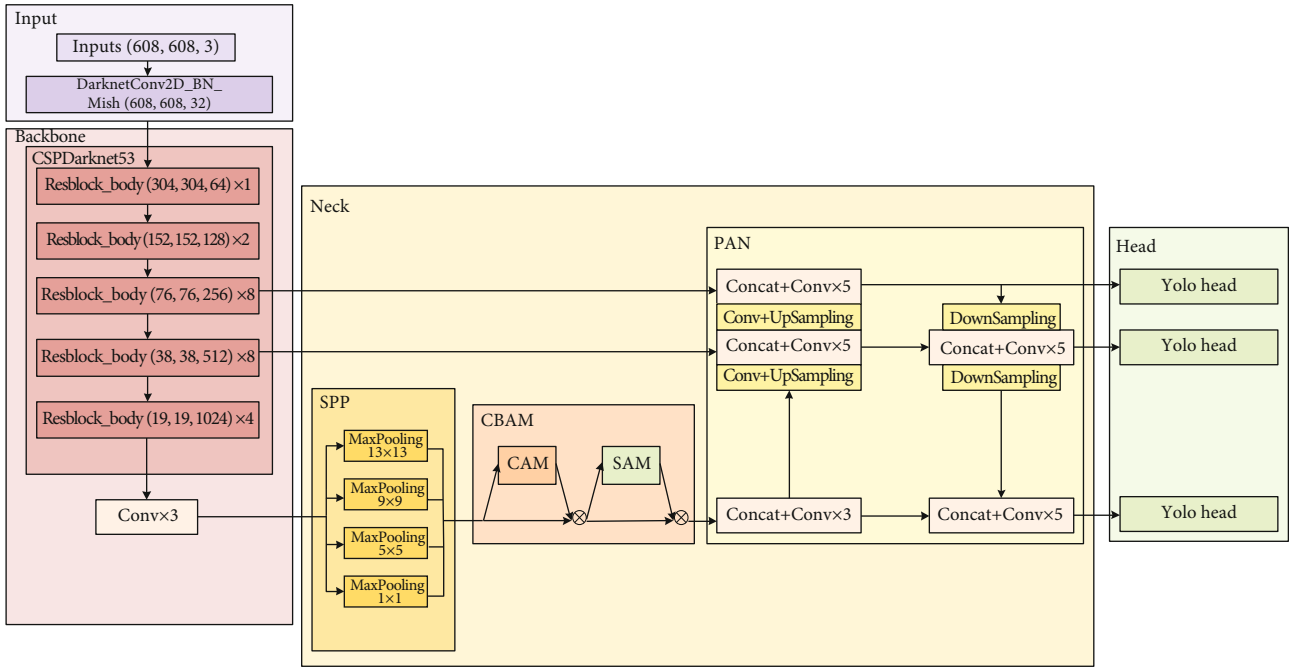


Figure 2: SPDNet network structure.

multiple scales ranging from cars to birds. It can be seen that the size difference of the original preset anchors is large, which has promising practicability in the object detection task under ordinary scenes. However, in the scene dominated by passengers in the subway station, the shorter and wider frames in the original anchors obviously have a poor matching effect on narrow and tall objects such as passengers, resulting in a lower performance of YOLOv4 detection head when calculating the IoU (intersection over union). In order to improve the utilization efficiency of anchors and get closer to the human body, this paper conducts a bounding box clustering analysis in advance for the target size in the dataset dominated by passengers.

In this paper, we use K-means++ [4] algorithm to perform clustering analysis on the objects in the KITTI dataset with a large number of passenger samples. K-means++ algorithm first randomly selects a sample as the initial clustering center, then calculates the distance $D(x)$ between each sample x_i and the existing clustering center point, and then calculates the probability $P(x)$ of each sample being selected as the next clustering center.

$$P(x) = \frac{D(x)}{\Sigma D(x)^2}. \tag{4}$$

Then, select the next clustering center through the roulette wheel selection method, and repeat the above steps to calculate the distance $D(x)$ and probability $P(x)$ until K anchors are obtained. Finally, the distance from each sample to K points in the clustering center is repeatedly calculated, and the sample point is divided into the class with the smallest distance to the clustering center, and then, the clustering center is updated until the size of the obtained anchors no longer changes.

Compared with the traditional K-means clustering algorithm, K-means++ optimizes the selection of the initial point and can significantly improve the error of the classification results, so as to obtain the anchors more suitable for the passenger and improve the accuracy of detection. In the case of dense crowds under the scene of a subway station, the missed detection caused by incorrect suppression of true positive box by postprocessing the NMS algorithm is well alleviated.

3.4. Attention Mechanism Model. SENet [40] first introduces an effective mechanism Squeeze-and-Excitation (SE) to learn channel attention and achieve encouraging results. This method uses global average pooling (GAP) [41], which is the squeeze operation, to obtain the global receptive field, and then through the global connection and nonlinear transformation, which is the excitation process, to explicitly construct the correlation of the characteristic channels. The author uses this structure to enhance the significant features and weaken the unimportant features, thereby making the extracted features more directional. However, the dimensionality reduction operation of this method in the fully connected layer will have a negative impact on channel attention, and it is unnecessary to use the fully connected layer to capture the correlation of all channels. Thus, in order to learn more channel features and reduce the complexity of the model, Wang et al. [42] proposes the Efficient Channel Attention (ECA) module. ECA adopts the method of avoiding dimensionality reduction and appropriate cross-channel interaction and has made a breakthrough improvement on the SE module through fast one-dimensional convolution to generate channel attention.

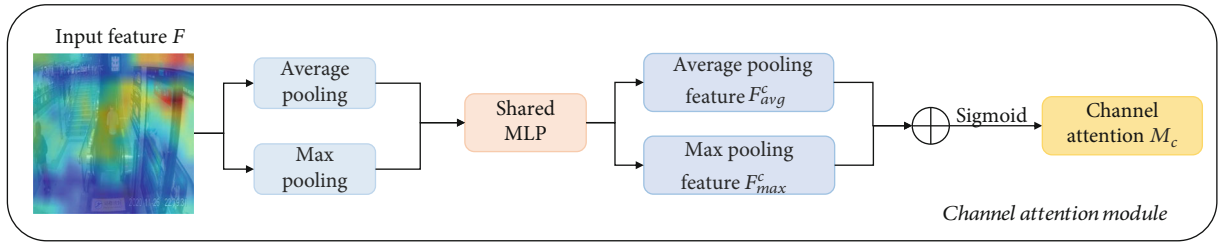


Figure 3: Channel attention module (CAM). The passenger feature map F in the middle layer is input, and after processing by the pooling layer and the shared fully connected layer, the passenger channel attention feature map M_C is obtained as the output. σ is the Sigmoid activation function.

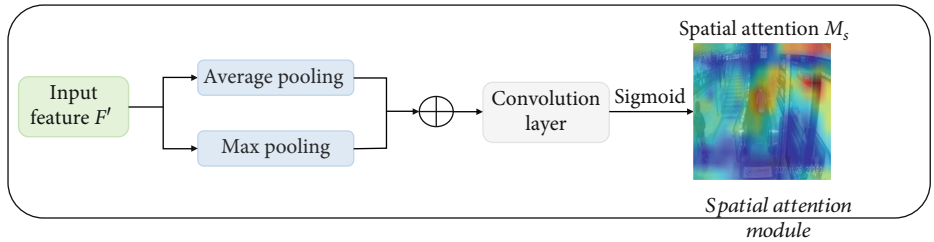


Figure 4: Spatial attention module (SAM). F obtained in Equation (5) is used as input, and the passenger's spatial attention feature map M_S after the pooling and convolution is used as the output. σ is the Sigmoid activation function.

However, in the object detection task, the target object is usually small, and the number of irrelevant objects is large. Therefore, the average value of the channel feature obtained after averaging pooling does not reflect the detection ability of the network very well, while the maximum pooling operation on channel features may better show the predictive ability of the model. Moreover, the SE and ECA modules only enhance the channel features in the feature map but lack the learning and enhancement of spatial features, so their ability to improve the model is limited.

In order to attain richer information, [3] introduces the Convolutional Block Attention Module (CBAM), an attention mechanism that can simultaneously focus on channel and spatial features. The channel domain attention adds weight to the signal on each channel, and the weight represents the degree of correlation between the channel and the crucial information of passengers. The spatial area attention can be understood as where the neural network is looking, which transforms the spatial domain information in the picture, so as to extract the main passenger information. At the same time, they also explore the placement of the channel and the spatial attention module and obtain the channel first and then the spatial attention mechanism module to be slightly better than other placement effects.

To gather more important information for the passenger detection task from images and reduce the interference of invalid feature information, we introduce the attention mechanism module CBAM. It combines the two-dimensional attention mechanism of feature channel and feature space, so that the spatial relationship of passenger features can supplement the position information that the channel attention mechanism cannot obtain. This reduces the influence of light conditions on the passenger detection task of subway stations and also improves the detection effect of the network on small-scale passengers. For the sake of clarity, given a passenger feature map $F \in R^{C\times H\times W}$ in the middle layer as input, where C, H, and W, respectively, represent the channel, height, and width of each passenger feature map. CBAM [3] inferred the one-dimensional channel attention feature map $M_C \in R^{C\times 1\times 1}$ and the two-dimensional spatial attention feature map $M_S \in R^{1\times H\times W}$ as follows:

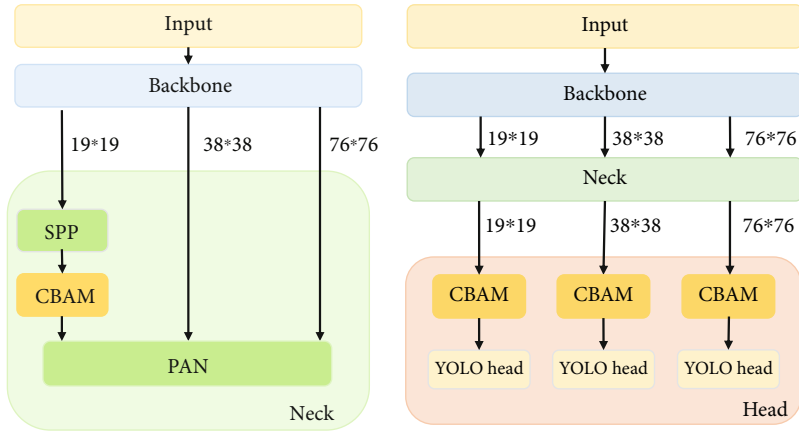
$$F' = M_C(F) \otimes F, \tag{5}$$

$$F'' = M_S\left(F'\right) \otimes F'. \tag{6}$$

$\otimes$ is the Hadamard product, which represents the product of corresponding elements in the matrix. First, multiply the channel attention feature map and the input feature map of passengers to get F', then calculate the spatial attention feature map of F', and finally, multiply the two to get the output F''.

3.4.1. Channel Attention Module. The channel attention mechanism is to model the dependence between the various channels of the passenger feature map, so as to pay more attention to what is meaningful in the input passenger image.

As shown in Figure 3, in order to aggregate the spatial features, CBAM obtains the average passenger feature information F^c_{avg} through average pooling of passenger feature information and obtains the most significant passenger feature F^c_{max} through the max pooling. Then, it uses a shared fully connected layer composed of multilayer perceptron to calculate the two different spatial background descriptions



(a) SPDNet-A (b) SPDNet-B

Figure 5: Two YOLOv4 models embedded in CBAM.

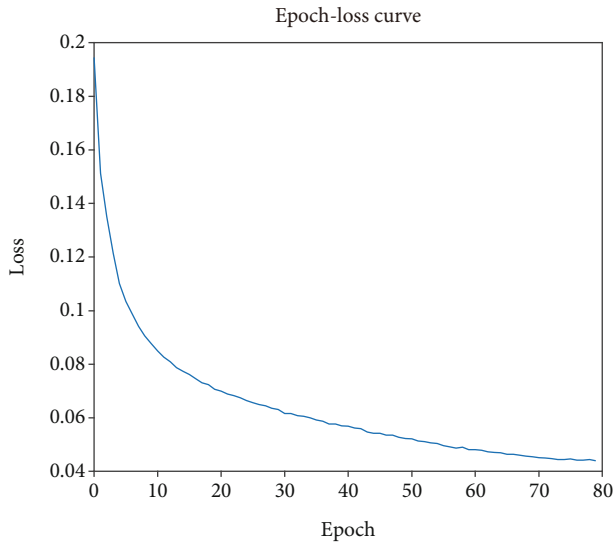


Figure 6: Loss function curve.

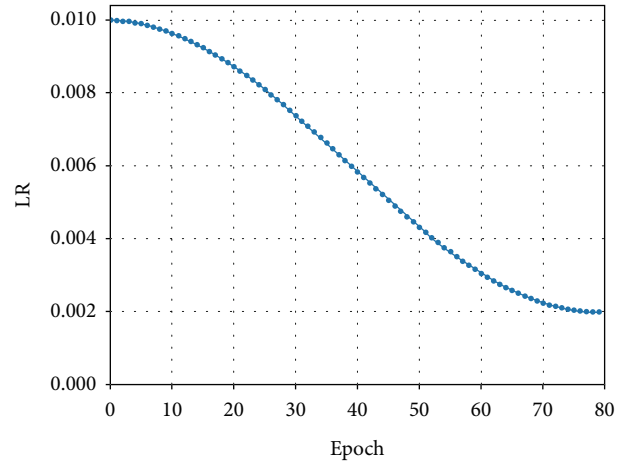


Figure 7: Learning rate curve.

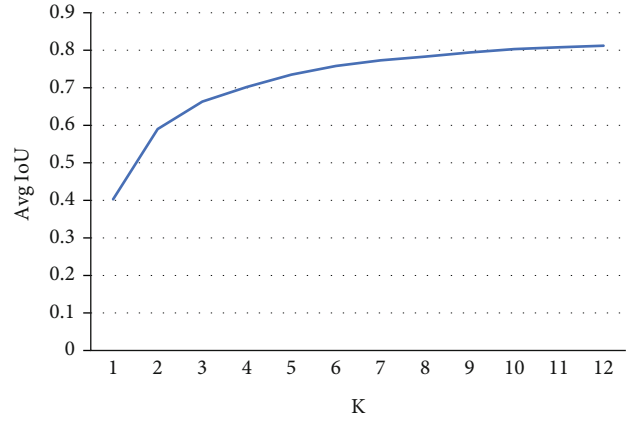


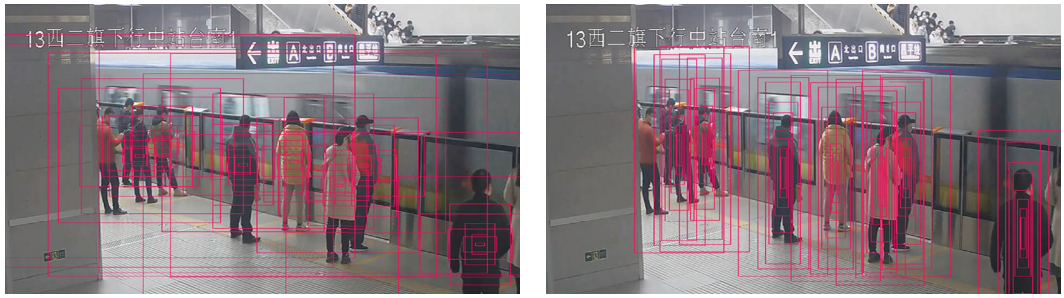
Figure 8: K-means++ algorithm clustering results.

and sum them element by element. Finally, the final channel attention feature map is generated through the activation function M_C:

$$M_C(F) = \sigma(\mathrm{MLP}(\mathrm{AvgPool}(F)) + \mathrm{MLP}(\mathrm{MaxPool}(F))) = \sigma\left(W_1\left(W_0\left(F^c_{\mathrm{avg}}\right)\right) + W_1(W_0(F^c_{\mathrm{max}}))\right). \quad (7)$$

Among them, σ is the Sigmoid activation function, W_0 is the first layer of the shared fully connected layer, the output vector length is $r \times C$, W_1 is the second layer of the shared fully connected layer, and the output vector length is C. The obtained channel attention feature map and the input map are subjected to elementwise multiplication operations to generate the input passenger characteristics required by the spatial attention module.

3.4.2. Spatial Attention Module. Spatial attention is used to accurately locate passenger features in space. The realization is shown in Figure 4. First, global average pooling and global maximum pooling are used to perform average and maximum operations on the input passenger features in the



(a) Before (b) After

Figure 9: Anchors before and after optimization comparison.

channel dimension, respectively. Then, the two passenger feature maps obtained are spliced and subjected to a convolution operation to reduce the dimension. It is one channel to ensure that the output passenger feature map is consistent with the input passenger feature map in the spatial dimension. The calculation formula of the spatial attention module M_S is shown in formula:

$$\begin{aligned} M_S(F) &= \sigma\left(f^{7\times7}([\text{AvgPool}(F);\text{MaxPool}(F)])\right) \\ &= \sigma\left(f^{7\times7}\left(\left[F^c_{\text{avg}};F^c_{\text{max}}\right]\right)\right). \end{aligned} \tag{8}$$

As a lightweight general module, CBAM can be integrated into the object detection network YOLOv4 for training. In this paper, CBAM automatically obtains the importance of each feature channel and feature space through learning, selects information related to passengers from a large amount of information, suppresses unimportant background information, and shows good passenger detection accuracy. We select the network model after adding the above three attention mechanisms to the YOLOv4 network as a comparison. The details are shown in Experiment 4.3.

3.4.3. Embedded Design of CBAM. In this paper, we discuss the performance of the CBAM embedded in different positions in the YOLOv4 network. Since the passenger feature map obtained after the backbone network has richer semantic features, and the model parameters after the pooling layer SPP are less; the calculation is more efficient, so we hope that the attention mechanism can further learn the features extracted from the backbone network.

Therefore, we put the CBAM into the deeper neck of the network, as shown in Figure 5(a), and put it into the detection head network, as shown in Figure 5(b), to obtain two new network models SPDNet-A and SPDNet-B.

4. Experiments

4.1. Dataset. We evaluate the object detection performance of our method on the KITTI [5] dataset. KITTI is one of the most important computer vision algorithm evaluation datasets. It has 7481 training images and 7518 test images, and the resolution of the images is roughly $1240 * 376$ pixels. This dataset judges the correctness of object positioning by comparing the degree of overlap (IoU) between the predicted box and the ground truth box with the size of the threshold and uses the mean Average Precision (mAP) to evaluate the results of the single-class target detection model. For each object, data such as its type and the location of the bounding box (including the four coordinates of up, down, left, and right) are marked. For the passenger detection, most of the pictures are a single target, a small part are two targets, and the rest are more targets.

Table 1: The quantitative results of different models on the KITTI dataset are compared.

Network model	mAP@0.5	Person (AP@0.5)
YOLOv4 (baseline)	86.3%	78.3%
SPDNet-A	84.4%	73.7%
SPDNet-B	89.1%	79.7%

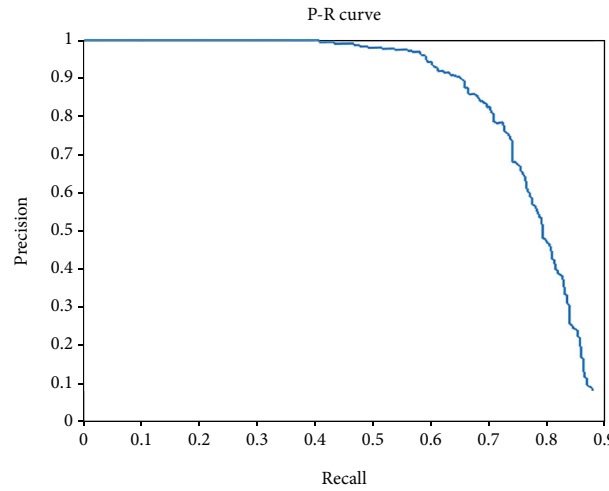


Figure 10: P-R curve.

The height of human bodies is normally distributed between 1.4 meters and 2.0 meters, and all angles are roughly evenly distributed.

We also collected video images of a subway station in Beijing and extracted frames to obtain images of passengers in the subway station with a resolution of $1920 * 1080$. In order to adapt to the network input, we adjust the picture

Table 2: Comparison of different attention methods on KITTI from the perspective of mean Average Precision (mAP), Person Average Precision (Person AP), Supplemental Module Parameters (Module Param), and Floating-Point Operations per second (FLOPs).

Network model	mAP@0.5	Person (AP@0.5)	Module Param	FLOPs
YOLOv4 (baseline)	86.3%	78.3%	0	128.46B
YOLOv4-ECA	87.7%	76.3%	5	128.46B + 0.13M
YOLOv4-SE	88.4%	76.8%	32768	128.46B + 0.16M
SPDNet	89.1%	79.7%	32771	128.46B + 0.20M

to an image with a resolution of 608 ∗ 608. In the subway station dataset, we collected passenger flow images of three scenes: gates, platforms, and escalators, including different camera angles, one-way passenger flow, good light and poor light, and passenger flow of different densities. Realize the multidirectional and comprehensive collection of passenger flow information.

4.2. Implementation Details. In our experiments, we randomly divide the training set and the validation set from the training set provided by the KITTI dataset at a ratio of 9 : 1. In particular, to better demonstrate the effect of our network on passenger detection, we separately test the data of the category of passengers. The training set and the test set have 6705, 775 images, respectively. In order to adapt to the input of the network, we adjust the size of the input picture to 640 ∗ 640.

The experimental configuration environment of this paper is as follows: CPU is IntelXeonE5-2678v3@2.5GHz, GPU is NVIDIA GeForce GTX 1080Ti. Using Python3.6.9, the deep learning framework is PyTorch1.6. The initial learning rate is set to 0.01, the cosine annealing method is used to adjust the learning rate, and the Adam optimization algorithm is used. The initial learning rate is 0.001, the weight decay is set to 0.0005, the epoch is 80, and the batch size is 16. Figures 6 and 7 show the details of network training.

4.3. The Result of Anchor Optimization. In the experiment, the K-means++ algorithm is used to carry out cluster analysis on the targets of the KITTI dataset, and the average intersection over union (Avg IoU) is used as the evaluation criterion for clustering. The results are shown in Figure 8, which represents the results of Avg IoU with K initial anchors.

In order to ensure the accuracy of prediction results and avoid huge computation caused by too many anchors, this experiment selects the anchors generated by clustering when $K = 9$ and Avg IoU = 79.38%. Finally, we cluster to generate anchors of size (6,57), (9,89), (14,124), (20,155), (24,217), (35,256), (42,327), (61,344), and (106,370). And Figure 9 shows how the anchors fit with passengers.

4.4. Ablation Study and Comparison Experiment. Table 1 shows that the model placed on the neck of the network can effectively improve the detection performance of YOLOv4. Compared with YOLOv4, mAP@0.5 is increased by 2.8%, and the Person (AP@0.5) is increased by 1.4%.

Table 3: Comparison of object detection results of different methods on KITTI from mean Average Precision (mAP calculated by Car and Person) and Person Average Precision (Person AP).

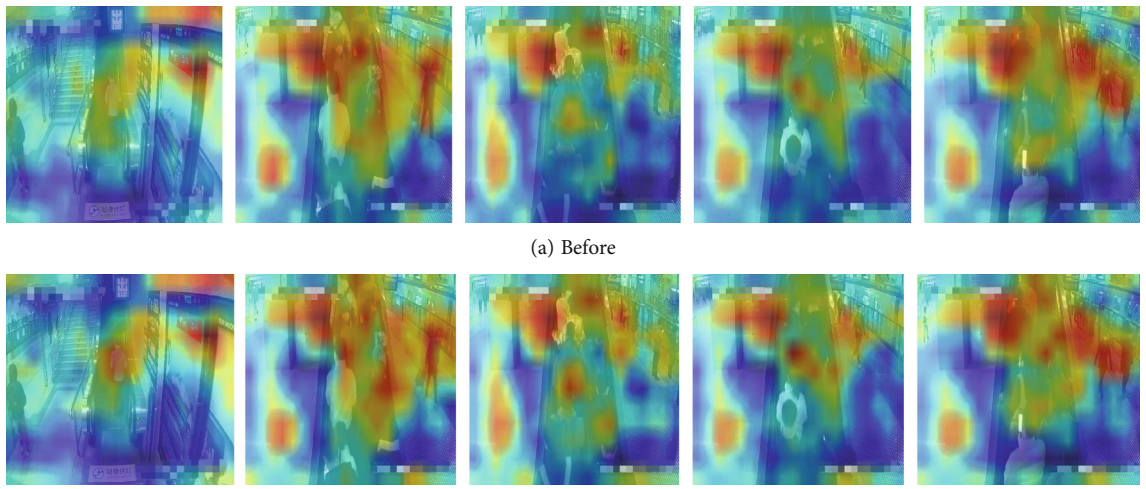
Network model	mAP@0.5	Person (AP@0.5)
Faster RCNN	62.3%	52.1%
SSD	63.8%	41.6%
YOLOv3	86.8%	77.9%
YOLOv4	86.3%	78.3%
SPDNet	87.5%	79.7%

The accuracy of the model with the mechanism on the head has decreased, and mAP@0.5 and Person (AP@0.5) have decreased by 1.9% and 4.6%, respectively.

The reason is that the feature map that reaches the head of the network is already a highly integrated feature map. Small targets such as passengers have become blurred and difficult to identify. At this time, embedding the attention mechanism module may lead to improper distribution of feature weights and ultimately reduce accuracy. However, the model placed on the neck of the network learns the features that have just been extracted by the backbone network, and the shallow information is retained. Therefore, the attention mechanism module can better enhance the spatial features and channel features to achieve higher detection results. In Figure 10, we also show the effect of our network through the P-R curve.

Our experiment also compares different attention mechanisms and YOLOv4 network integration on KITTI. The specific results are shown in Table 2.

Table 2 shows that the mAP@0.5 and Person (AP@0.5) of the YOLOv4 network on the KITTI dataset are 86.3% and 78.3%. After introducing the attention mechanisms ECA and SE into the YOLOv4 network, the performance of the network model has been improved to varying degrees, but their ability to detect passengers has decreased. The CBAM module has the best effect on improving the performance of the network. When the parameter amount is similar to the SE, we not only improve the Person (AP@0.5), but the accuracy of detecting other categories has also been greatly improved. Also, the index Frames Per Second (Frames Per Second, FPS), which measures the complexity of our network, is 55.2, which shows that our network has good real-time performance. The reason is that SE and ECA only use the channel attention mechanism to focus on meaningful features, and only use global average



(a) Before

(b) After

Figure 11: Heat maps before and after attention module comparison.

(a) Escalator1 (b) Escalator2

(c) Gate1 (d) Gate2

(e) Gate3 (f) Platform1

(g) Platform2 (h) platform1

Figure 12: Practical application effects.

pooling; CBAM uses global average pooling and maximum pooling to summarize spatial features and also pays attention to where the features are meaningful.

Table 2 can show that in the YOLOv4 network, adding the proposed channel attention mechanism and spatial attention mechanism can effectively improve the detection performance of the network.

The results in Table 3 show that our network model has reached 79.7% of passenger detection accuracy and 87.5% of mAP@0.5. Compared with other methods in

Table 3, our detection accuracy has been significantly improved.

4.5. Transfer Learning and Application. In this section, we will show the test images of the subway station passenger dataset and the effect images processed by our network. The SPDNet method proposed in this paper is to solve the problems of low detection accuracy and high missed detection rate of subway passengers. In the absence of datasets, the model is first trained in the KITTI dataset, and then, the transfer learning method is adopted to fine-tune the model. Through transfer learning, we can share the learned model parameters with the new model, thereby speeding up and optimizing the learning efficiency of the model instead of learning from zero. This makes the network more suitable for passenger detection in subway scenarios. Experiments show that the method proposed in this paper is effective. Figure 11 shows that after the introduction of attention mechanism, passenger characteristics are more concentrated, and Figure 12 shows the real situation of subway station passenger detection.

5. Conclusions

This research is aimed at realizing a real-time detection method of subway station passengers based on surveillance systems, so as to better serve production and life. We propose a network model SPDNet (Subway Passengers Detection Net) based on YOLOv4. In this model, an attention mechanism module is introduced to calibrate and fuse channel features in dimension and spatial dimension.

Feature relations in spatial dimension supplement target location information missing in channel dimension and improve the capture ability of small passengers caused by the angle of surveillance camera in the station. At the same time, the attention mechanism module also makes the network focus, enhances the passenger characteristics, and restrains the interference of the subway station light conditions. We have carried out sufficient comparative tests and ablation experiments to verify the effectiveness of our method. In the meantime, we use the K-means++ algorithm to generate anchors that are more consistent with the passenger aspect ratio based on the dataset KITTI, which increases utilization of anchor box and mitigates the missing caused by the incorrect suppression of true positive boxes by the Nonmaximum Suppression algorithm.

Compared with the original YOLOv4, SPDNet improves the mAP@0.5 and Person (AP@0.5) by 2.8% and 1.4%, respectively. The speed of detection is up to 55.2 FPS, which also meets the real-time requirements.

Finally, based on the technology of transfer learning, we apply our network 575 to the actual scene of the subway station and achieve inspiring results, which better eliminates the interference of light conditions and monitoring equipment angle in the subway station on passenger detection task and has achieved good engineering effect. The problem-solving method adopted in this paper can be transferred to the passenger detection task in other similar scenes, and it also has certain guiding significance for the research on the migration of several other attention mechanisms to the one-stage detector algorithm.

In this paper, the K-means++ algorithm is adopted to reduce the leak detection caused by occlusion in crowded areas, but its mitigation effect is limited. In the next step, the passenger detection problem under the condition of shelter in subway stations will be further studied in order to create a simple and efficient method to solve the problem and further realize strong intelligent detection.

Data Availability

We first train and test the model on the KITTI dataset (http://www.cvlibs.net/datasets/kitti/index.php) and then transfer the model to the subway pedestrian dataset collected and annotated by ourselves to adapt to the subway station scene. The subway pedestrian dataset is not public.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported in part by the Beijing Municipal Natural Science Foundation (L201015) and in part by the National Key R&D Program of China (2020YFC0833104).

References

[1] J. Yang, X. Dong, and S. Jin, "Metro passenger flow prediction model using attention-based neural network," *IEEE Access*, vol. 8, pp. 30953–30959, 2020.

[2] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: optimal speed and accuracy of object detection," 2020, https://arxiv.org/abs/2004.10934.

[3] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, Munich, Germany, 2018, https://eccv2018.org/.

[4] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii, "Scalable k-means++," 2012, https://arxiv.org/abs/1203.6402.

[5] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: the Kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[6] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: a survey," 2019, https://arxiv.org/abs/1905.05055.

[7] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, Kauai, HI, USA, 2001.

[8] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *2008 IEEE conference on computer vision and pattern recognition*, pp. 1–8, Anchorage, AK, USA, 2008.

[9] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *2010 IEEE Computer society conference on computer vision and pattern recognition*, pp. 2241–2248, San Francisco, CA, USA, 2010.

[10] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[11] R. Girshick, P. Felzenszwalb, and D. McAllester, "Object detection with grammar models," *Advances in Neural Information Processing Systems*, vol. 24, pp. 442–450, 2011.

[12] R. B. Girshick, *From Rigid Templates to Grammars: Object Detection with Structured Models*, Citeseer, 2012.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, Columbus, Ohio, 2014, http://www.pamitc.org/cvpr14/.

[15] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, Santiago, Chile, 2015, https://www.aconf.org/conf_58344.html.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," 2015, https://arxiv.org/abs/1506.01497.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[18] K. He, G. Gkioxari, P. Doll'ar, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, Italy, 2017, https://www.aconf.org/conf_108682.html.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, Las Vegas, Nevada, 2016, https://cvpr2016.thecvf.com/.

[20] W. Liu, D. Anguelov, D. Erhan et al., "Ssd: single shot multibox detector," in *European conference on computer vision*, Springer, Cham, 2016.

[21] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271, Honolulu, Hawaii, 2017, https://cvpr2017.thecvf.com/.

[22] J. Redmon and A. Farhadi, "Yolov3: an incremental improvement," 2018, https://arxiv.org/abs/1804.02767.

[23] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "Dssd: deconvolutional single shot detector," 2017, https://arxiv.org/abs/1701.06659.

[24] Z. Li and F. Zhou, "Fssd: feature fusion single shot multibox detector," 2017, https://arxiv.org/abs/1712.00960.

[25] J. A. Antonio and M. Romero, "Pedestrians' detection methods in video images: a literature review," in *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 354–360, Las Vegas, NV, USA, 2018.

[26] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, pp. 886–893, San Diego, CA, USA, 2005.

[27] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1904–1912, Santiago, Chile, 2015, https://www.aconf.org/conf_58344.html.

[28] J. Li, X. Liang, S. Shen, T. Xu, J. Feng, and S. Yan, "Scale-aware fast r-cnn for pedestrian detection," *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 1–996, 2017.

[29] Y. Lu, T. Javidi, and S. Lazebnik, "Adaptive object detection using adjacency and zoom prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2351–2359, Las Vegas, Nevada, 2016, https://cvpr2016.thecvf.com/.

[30] X. Sun, P. Wu, and S. C. Hoi, "Face detection using deep learning: an improved faster rcnn approach," *Neurocomputing*, vol. 299, pp. 42–50, 2018.

[31] L. Zhu, H. Liang, H. Wang, B. Ning, and T. Tang, "Joint security and train control design in blockchain empowered cbtc system," *IEEE Internet of Things Journal*, no. 99, p. 1, 2021.

[32] L. Zhu, Y. Li, F. R. Yu, B. Ning, T. Tang, and X. Wang, "Cross-layer defense methods for jamming-resistant cbtc systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, 2021.

[33] Y. Li, L. Zhu, H. Wang, F. R. Yu, and S. Liu, "A cross-layer defense scheme for edge intelligence-enabled cbtc systems against mitm attacks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2286–2298, 2021.

[34] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, Las Vegas, Nevada, 2015, https://cvpr2016.thecvf.com/.

[35] T.-Y. Lin, P. Doll'ar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, Boston, Massachusetts, 2017, https://cvpr2015.thecvf.com/.

[36] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European conference on computer vision*, Springer, 2016.

[37] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "Cspnet: a new backbone that can enhance learning capability of cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 390-391, 2020, https://cvpr2020.thecvf.com/.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[39] Y. Mei, Y. Fan, Y. Zhang et al., "Pyramid attention networks for image restoration," 2020, https://arxiv.org/abs/2004.13824.

[40] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, Salt Lake City, Utah, 2018, https://cvpr2018.thecvf.com/.

[41] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, https://arxiv.org/abs/1312.4400.

[42] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: efficient channel attention for deep convolutional neural networks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11531–11539, Seattle, WA, USA, 2020.