Noname manuscript No.
(will be inserted by the editor)

From macro to micro: Rethinking multi-scale pedestrian detection

Yuzhe He · Ning He · Haigang Yu · Ren Zhang · Kang Yan

Received: date / Accepted: date

Abstract Pedestrian detection is the use of computer vision techniques to determine whether there are pedestrians in an image or video sequence and give their precise positioning, but the difference in the scale of pedestrians has always been a difficult problem in pedestrian detection. In contrast to existing research, this study jointly considers the problem of multi-scale pedestrian detection at both the macro and micro levels. At the macro level, the shape and location of an anchor are predicted by feature maps to guide its generation, and the obtained anchor can better adapt to pedestrian targets at different scales. At the micro level, the standard convolution in the backbone network is replaced with switchable atrous convolution, which effectively solves the problem of scale differences between pedestrians. Finally, the classification and regression tasks in pedestrian detection are completed more efficiently through the use of a Double Head. These elements are combined to form a multi-scale pedestrian detection network, and experimental results show that the model proposed in this paper can substantially improve the performance of multi-scale pedestrian detection. The detection accuracy on the COCOPersons dataset reaches an average precision (AP) of 57.3. Compared with the pedestrian detection accuracy of Faster R-CNN based on a feature pyramid

✉Ning He
E-mail: xxthening@buu.edu.cn

Yuzhe He
E-mail: 849836043@qq.com

Haigang Yu
E-mail: 1184555290@qq.com

Ren Zhang
E-mail: 851769359@qq.com

Kang Yan
E-mail: 1399471158@qq.com
Beijing Union University, Beijing 100101, China

network at large, medium, and small scales, the accuracy of our model is significantly improved at 1.7 AP, 2.5 AP, and 6.8 AP, respectively. On the Caltech pedestrian dataset, the MR^2 of Near, Medium and Far subsets reach 0.45%, 13.78% and 48.85%, respectively. And on the CityPersons pedestrian dataset, the MR^2 of Small, Medium and Large subsets reach 12.1%, 2.6% and 5.5%, respectively.

Keywords Macro level · Micro level · Pedestrian detection · Scale difference

1 Introduction

As the field of deep learning algorithm research matures, the problem of object detection in computer vision has attracted the attention of an increasing number of researchers. As an important sub-topic in computer vision, pedestrian detection has a very wide range of application scenarios, such as autonomous driving, intelligent monitoring, intelligent robots, etc. It plays an extremely important role in intelligent monitoring. In complex application scenarios, there are differences in the pedestrian scale displayed by the monitoring due to the distance of the pedestrian target from the intelligent monitoring device. For example, pedestrian objects that are closer to the monitoring device tend to occupy more pixels in the image, and pedestrian objects that are farther away from the monitoring device tend to occupy less area in the image. This leads to, on the one hand, the small-scale pedestrian targets have blurred outlines and less useful information, making it difficult for detectors to accurately detect them. On the other hand, the feature of large-scale pedestrian targets and small-scale pedestrian targets are quite different, and it is difficult to design a unified feature processing strategy for targets of different scales. Therefore, it is still a great challenge to detect pedestrian objects of different scales under intelligent surveillance equipment.

Pedestrian detection methods can be subdivided into two categories: traditional pedestrian detection methods and deep learning-based pedestrian detection methods. Traditional pedestrian detection methods need to give pedestrian features before detection, and then match and identify by defining pedestrian texture and gradient features. The more typical ones include methods based on the Local Binary Pattern (LBP) [35] and the Histogram of Oriented Gradient (HOG) [11]. However, due to the simple extraction of manual features and imperfect post-processing, in complex scenes, the pedestrian detection method based on deep learning is more dominant. Because convolutional neural networks are widely used in image classification problems, deep learning methods can automatically extract better pedestrian features while learning to obtain better similarity measures, and hence the accuracy of pedestrian detection techniques has been greatly improved. Pedestrian detection methods based on deep learning regard pedestrians as a specific target and adopt general target detection methods, including one-stage detectors that dominate speed and two-stage detectors that dominate in accuracy. Pedestrian detection

uses computer vision techniques to determine the presence and precise positioning of pedestrians in an image or video sequence. However, scale differences between individual pedestrians continue to hinder the performance of general detectors based on deep learning. In deep learning-based detector, the image features are extracted through the standard convolution in the backbone network, and the traditional anchor mechanism is used to obtain the proposals for detection. In fact, this method is not the best approach to this problem, and hence this paper discusses the problems at the macro and micro levels separately.

At the macro level, we consider that the traditional anchor mechanism is not suitable for solving the problem of multi-scale pedestrian detection. The most advanced detectors currently use a traditional anchor mechanism, that is, a set of anchors with a defined shape and size are uniformly placed on the image. This causes two problems: (1) A set of anchors with a fixed size and aspect ratio must be predefined. However, the scale of pedestrian targets in pedestrian detection is quite different, and this is not conducive to detection. Incorrect anchor design may affect the speed and accuracy of the detector. (2) To maintain a sufficiently high recall, a large number of anchors are required. However, most anchors have nothing to do with the object of interest (many anchors are in the background area). Moreover, a large number of anchors will increase the computational cost. Therefore, we propose an adaptive anchor mechanism to predict the position and shape of the anchor through the feature maps to guide its generation. This solves the above problems well and is helpful for subsequent multi-scale pedestrian target detection.

At the micro level, we found that standard convolution and atrous convolution cannot adapt well to pedestrian targets of different scales because of the single receptive field. The current mainstream backbone network uses a large number of 3×3 standard convolutions for feature extraction, but the difference in pedestrian target scales in pedestrian detection is obvious, and the single standard convolution receptive field limits multi-scale pedestrian detection. Moreover, standard convolution reduces the size of the image by pooling while increasing the receptive field of the network. Finally, the upsampling operation is used to restore the size of the image to the original size. Such a series of operations leads to the loss of some information of the image, especially the detailed information that may have a relatively large impact on performance. To increase the receptive field without losing image information, in 2015, Yu et al. [52] proposed atrous convolution. Atrous convolution is an effective technique that can amplify the filter's receptive field in any convolutional layer. However, some problems with atrous convolution remain. The specific performance is: the information obtained from a long distance has no relevance. Because of the sparsely sampled input signal of the atrous convolution, there is no correlation between the information obtained by the long-distance convolution, which affects the detection result. Atrous convolution obtains long-distance information by increasing the atrous rate to expand the receptive field. However, the feature information extracted by atrous convolution with a large atrous rate may be effective for some larger

targets, but not good for small targets. Therefore, by switching the atrous rate of the atrous convolution, we hope to use atrous convolution with a larger atrous rate for larger-scale pedestrian targets and atrous convolution with a smaller atrous rate for smaller-scale pedestrian targets.

The main contributions of this paper are as follows: (1) We systematically discuss why the existing detection methods at the macro and micro levels are not suitable for solving the problem of scale differences that are common in pedestrian detection. (2) We propose a multi-scale pedestrian detection model to solve the problem of pedestrian detection scale differences from multiple angles. At the macro level, the guided anchoring region proposal network (GARPN) is used to predict the position and shape of an anchor based on the feature maps, and the anchor is generated in an adaptive manner. This scheme can handle pedestrian targets at different scales better than the traditional anchor scheme. At the micro level, standard convolution in the backbone network is replaced with switchable atrous convolution (SAC). According to the target position of pedestrians of different scales, we switch atrous convolution with different atrous rates. Finally, we use a Double Head [49] to complete classification and regression tasks more efficiently. (3) This study evaluated the proposed model on the COCOPersons dataset [28] and compared it with the current mainstream detection models and other multi-scale methods. Ablation experiments on each module of the proposed model were also conducted in this study, and the effect of each component of the SAC on performance was evaluated. At the same time, this paper also verifies the experimental results of the proposed model on the Caltech [13] and CityPersons pedestrian datasets [56].

2 Related work

This section first introduces the multi-scale pedestrian detection methods and summarizes related work at the macro and micro levels with respect to scale difference in pedestrian detection. At the macro level, the use of a traditional anchor mechanism and its improved methods are reviewed. At the micro level, the related work of atrous convolution is presented.

The common multi-scale pedestrian detection methods are multi-scale feature fusion methods and anchor-free methods.Multi-scale feature fusion pedestrian detection methods, such as in 2017, Zhu et al. [60] proposed SADR, introducing deconvolution layer to adaptively up-sample the feature map of small pedestrians. In the same year, Du et al. [14] proposed F-DNN, which uses SSD to generate candidate pedestrians and uses a soft rejection strategy to fuse multiple DNNS in parallel to detect pedestrians. In 2021, Tan et al. [42] proposed the Bidirectional Feature Enhancement Module (BFEM), which enhances the semantic information of low-level features and enriches the positioning information of high-level features. Multi scale pedestrian detection method based on anchor free, such as in 2019, Liu et al. [30] proposed CSP. In 2020, Wang et al. [47] further refined the CSP. In the same year, Cai et al. [3] proposed PP-Net, an anchor-free method for center-based pedestrian

detection. Different from the above methods, this paper solves the problem of pedestrian detection scale difference from the macro and micro levels. At the macro level, we found that the traditional anchor frame mechanism and its improvement used in most pedestrian detection methods are not suitable for multi-scale pedestrian detection; At the micro level, we found that standard convolution and atrous convolution cannot adapt well to pedestrian targets of different scales because of the single receptive field.

At the macro level. The traditional anchor mechanism uses a sliding window method to generate proposals in the feature map, and it has been widely adopted by various anchor-based detectors. In 2015, Ren et al. [38] used a region proposal network in Faster R-CNN to generate target proposals. This network uses a fully convolutional network to map each sliding window anchor to a low-dimensional feature. Because of the alignment and consistency of the anchor mechanism, a method of using multi-scale anchors to handle objects of different scales has emerged. In 2016, Liu et al. [29] proposed the single shot multibox detector (SSD), which uses anchor regression to detect objects in multiple feature maps. In 2017, Lin et al. [26] proposed feature pyramid networks (FPN), which generate multi-level and multi-scale feature maps, and set anchors at one scale and three aspect ratios at each level. This network provides a solution to the problem of scale difference in pedestrian detection. In 2018, Lin et al. [27] proposed RetinaNet, which uses an anchor setting that is different from that of FPN. It sets anchors at three scales and three aspect ratios at each level of the feature maps. In the same year, Yang et al. [51] proposed MetaAnchor, which randomly samples anchors of any shape during training to cover different types of target bounding boxes. At the same time, the parameters remain unchanged. In 2020, Zhong et al. [59]proposed a general approach to optimize anchor boxes for object detection. To improve the accuracy and reduce the workload of designing anchor boxes, it is proposed to dynamically learn the anchor shape, which enables the anchors to automatically adapt to the data distribution and network learning ability. In the same year, Ma et al. [33] studied the problem of automatically optimizing anchor boxes for object detection. In 2021, Ming et al. [34] proposed a dynamic anchor learning (DAL) method, which utilizes the newly defined matching degree to comprehensively evaluate the localization potential of the anchors and carries out a more efficient label assignment process. In 2022, Liu et al. [31] proposed a feature-guided anchor generation method named dynamic anchor. Compared with the hand-designed anchor scheme, dynamic anchor discards all pre-defined boxes and avoids complex hyper-parameters. Unlike the above method, our proposed method uses GARPN [46] to predict the shape and position of the anchor through the feature map to guide its generation so that the obtained anchor can better adapt to pedestrian targets at different scales. Specifically: on the one hand, by predicting the position of the anchor, the possibility of the anchor appearing in the non-interesting area and the background area is greatly reduced. On the other hand, by predicting the shape of the anchor, the obtained anchor can be more suitable for pedestrian targets with large scale differences than the traditional anchor mechanism.

At the micro level. Researchers have found that using atrous convolution is more conducive to multi-scale detection than standard convolution. In 2017, Yu et al. [53] combined a residual network with atrous convolution, and proved through experiments that the performance of the atrous residual network is much higher than an ordinary residual network. In the same year, Chen et al. [6] embedded the atrous spatial pyramid pooling (ASPP) module in DeepLabv3. This module is based on atrous convolution and spatial pyramid pooling. The given input is sampled in parallel by atrous convolution with different atrous rates. This is equivalent to capturing the context of the image at multiple scales, which helps improve the accuracy of detecting objects of different scales in the image. In 2018, Li et al. [25] replaced the original 3×3 convolutions in the bottleneck of a deep network with atrous convolutions with an atrous rate of 2. Without reducing the size of the space, this increases only slightly the amount of calculation while increasing the receptive field, thereby increasing the accuracy of detection. In 2020, Alsaih et al. [1] discussed the performance difference between convolution operations and atrous convolution operations. In 2021, Wang et al. [48] solved the grid effect by smoothing the atrous convolution itself instead of stacking atrous convolution layers. The grid effect refers to the inability to calculate all pixels of the feature map when stacking multiple atrous convolutions, thus losing the continuity of information. In the same year, Kim et al. [24] proposed an attention-based multi-scale atrous convolutional neural network (AMSASeg). And through distinctive atrous spatial pyramid pooling (DASPP) utilizes average pooling operations and atrous convolutions with different sizes to aggregate distinctive information on objects at multiple scales. Unlike the above method, We convert standard convolutions in the backbone network to SAC [37] and do not change the pretrained model in the process. Through the switch function, the atrous rate of the atrous convolution is controlled according to the pedestrian target scale, and the problem of the scale difference between pedestrians is effectively solved.

3 Methodology

To address the problem of scale differences in pedestrian detection, this paper considers the macro and micro levels separately, and proposes a multi-scale pedestrian detection network. At the macro level, considering that the traditional anchor mechanism is not suitable for solving the detection of pedestrian targets at different scales, an anchor adaptive mechanism is used to guide the generation of anchors. At the micro level, considering that standard convolution and atrous convolution have a single receptive field, which does not meet the requirements of multi-scale pedestrian detection, we proposed replacing the standard convolution in the backbone network with SAC. Finally, we reconsider the classification and regression tasks to further improve the performance of the detector. The overall structure of the proposed method is shown in Figure 1.

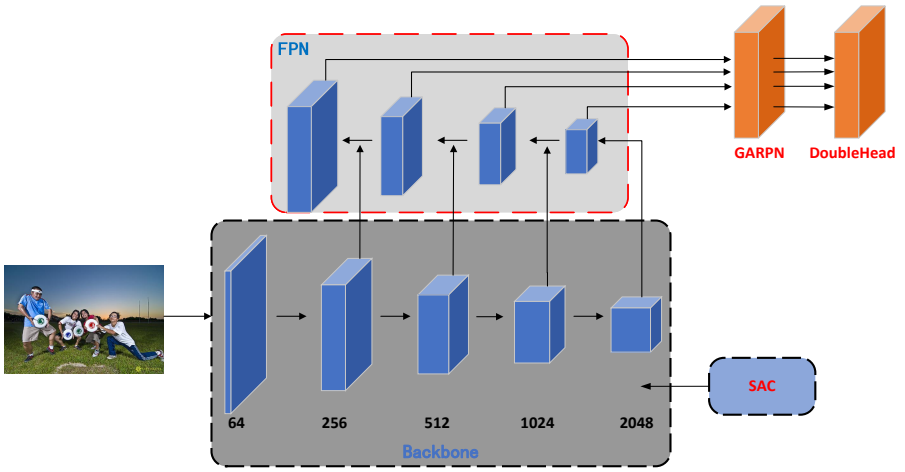

Fig. 1 Overall structure of the proposed method

As shown in Figure 1, we selected ResNet50 [19] as the backbone network of the model to extract the feature map of the image. The structure of ResNet50 is presented in Table 1. We replace all 3×3 convolutions with switchable atrous convolutions and switch the atrous convolution rate according to the pedestrian positions at different scales. First, a 7×7 convolution layer with a stride of two and a 3×3 maximum pooling layer with a stride of two greatly reduce the required storage space. Then, the data pass through a stack of residual blocks of 3, 4, 6, and 3 filters and 1×1 convolution. This reduces the dimensionality and restores the number of channels, which effectively reduces the computational complexity. The four residual blocks result in feature maps with channel numbers of 256, 512, 1024, and 2048, respectively. These feature maps extracted from the four groups of residual blocks are used to construct the feature pyramid to generate multi-scale feature expressions. Then, the multi-level feature maps are fed to the GARPN to predict the position and shape of the anchor to better adapt to pedestrian targets at different scales. Finally, the classification and regression tasks are completed through a Double Head.

3.1 GARPN

To better detect multi-scale pedestrians, at the macro level, the proposed method uses GARPN to predict the position and shape of the anchors through the feature maps to guide their generation. Its structure is shown in Figure 2. It consists of an anchor generation module and a feature adaptation module. The anchor generation module predicts the position and shape of an anchor through the N_L and N_S branches, respectively. The scheme is described as follows: the position and shape of a pedestrian target can be represented by a four-dimensional vector (x, y, w, h), where (x, y) is the center point of the space

Table 1 ResNet50 network structure

Layer name	Output size	50-layer
Conv 1	112×112	$7 \times 7, 64, stride2$
Conv2_x	56×56	$3 \times 3 maxpool, stride2$
		$\begin{bmatrix} 1*1,64 \\ 3*3,64 \\ 1*1,256 \end{bmatrix} * 3$
Conv3_x	28×28	$\begin{bmatrix} 1*1,128 \\ 3*3,128 \\ 1*1,512 \end{bmatrix} * 4$
Conv4_x	14×14	$\begin{bmatrix} 1*1,256 \\ 3*3,256 \\ 1*1,1024 \end{bmatrix} * 6$
Conv5_x	7×7	$\begin{bmatrix} 1*1,512 \\ 3*3,512 \\ 1*1,2048 \end{bmatrix} * 3$

coordinate, w represents the width of the bounding box, and h represents the height of the bounding box. Assuming that a target is extracted from image I, its position and shape are distributed as follows:

$$p(x, y, w, h|I) = p(x, y|I)p(w, h|x, y, I) \tag{1}$$

Equation (1) shows two important pieces of information: (1) Given an image, the target may only exist in certain areas; (2) the shape (size and aspect ratio) of a pedestrian target has a strong correlation with its location.

The anchor position prediction branch N_L is used to predict the anchor position, as shown in Figure 2. First, the N_L branch generates a probability map $p(\cdot|F_I)$ that is equal in size to the input feature map F_I, and each $p(i, j|F_I)$ corresponds to the position of the coordinate $((i+\frac{1}{2})s+(j+\frac{1}{2})s)$ on I, where s is the stride of the feature map, that is, the distance between adjacent anchors. The value of $p(i, j|F_I)$ represents the probability that the target center exists at that location. Probability mapping $p(i, j|F_I)$ uses a 1×1 convolution on F_I to obtain the mapping of the target score map, and then uses the sigmoid function to convert it into a probability value. Based on the generated probability map, we determine the active area where the target may exist by selecting a location with a probability value higher than a predefined threshold of ϵ_L. This process can filter out the vast majority of non-target areas while maintaining the same recall. As shown in Figure 6(b), the background area is excluded, and the anchors are all concentrated around the person.

After determining the possible locations of pedestrian targets, we use anchor shape prediction branch N_S to determine the shape of pedestrian targets at each location, as shown in Figure 2. Given a feature map F_I, the N_S branch predicts the best shape a for each position through a 1×1 convolution layer,

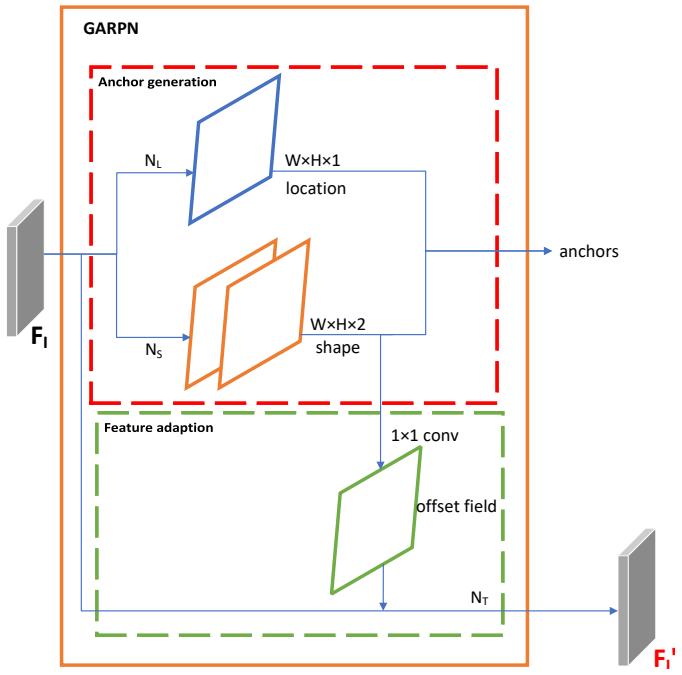

Fig. 2 GARPN structure

that is, the shape that produces the highest coverage with the nearest ground truth. Although our goal is to predict the values of width w and height h, based on experience, it has been found that direct prediction of these two values is not stable because they have a large range. Therefore, the conversion is performed as follows:

$$w = \sigma \cdot s \cdot e^{dw}, h = \sigma \cdot s \cdot e^{dh} \tag{2}$$

The shape prediction branch N_S outputs dw and dh, and we then map them to (w, h), where s is the stride and σ is an empirical scale factor ($\sigma = 8$ in the experiment). The nonlinear transformation mapping can map $[0, 1000]$ to $[-1, 1]$, making the learning target simpler and more stable. Experiments show that because of the close relationship between position and shape, our scheme can achieve higher recall than the traditional anchor mechanism. Because it allows any aspect ratio, the scheme can better fit pedestrian targets at different scales.

In the traditional anchor mechanism using the sliding window scheme, the anchors are consistent across the entire feature map, that is, they have the same size and aspect ratio at each position. Therefore, the traditional anchor mechanism uses a consistent full convolutional integrator on all feature maps. However, in the method proposed in this paper, the shape of the anchor varies from position to position, that is, the shape of the anchor is variable. Our aim is that the features of larger anchors can encode the content of larger areas,

and the features of smaller anchors can extract the content of smaller areas. On the basis of this, we introduce the feature adaptation module that transforms features according to the anchor shape at a specific position, as follows:

$$\boldsymbol{f_i^{'}} = N_T(\boldsymbol{f_i}, w_i, h_i) \tag{3}$$

Here, $\boldsymbol{f_i}$ is the feature of the i^{th} position and (w_i, h_i) is the corresponding anchor shape. For this position-dependent conversion method, we use a 3×3 deformable convolution layer [9] to implement N_T. As shown in Figure 2, the offset is first predicted from the output of the anchor shape prediction branch, and then a deformable convolution kernel offset is used on the original feature map to obtain $\boldsymbol{f_i^{'}}$. Using the adaptive features, we can perform further classification and bounding box regression more efficiently.

The method proposed in this paper uses multi-task loss to optimize in an end-to-end manner. In addition to the traditional classification loss L_{cls} and regression loss L_{reg}, we introduce two additional losses, namely the anchor position loss L_{loc} and anchor box shape loss L_{shape}. They are jointly optimized by the following equation.

$$L = \lambda_1 L_{loc} + \lambda_2 L_{shape} + L_{cls} + L_{reg} \tag{4}$$

To train the anchor position branch, for each image, we need a binary label map, where 1 represents an effective position to place the anchor and the other pixels are 0. In this work, we use the ground truth to guide the generation of binary label maps. Our aim is to place more anchors near the center of the target and fewer anchors away from the center. First, we map the ground truth bounding box (x_g, y_g, w_g, h_g) to the feature map scale to obtain $(x_g^{'}, y_g^{'}, w_g^{'}, h_g^{'})$. Here, $R(x, y, w, h)$ is a rectangular area with a center of (x, y) and a size of $w \times h$. We would like for more anchors to appear near the center of the ground truth target to obtain a larger IOU, and hence three types of regions are defined for each box:

(1) Central region $CR = R(x_g^{'}, y_g^{'}, \sigma_1 w^{'}, \sigma_1 h^{'})$ defines the central area of the box, and the pixels in CR are designated as positive samples, as indicated by the green area in Figure 3.

(2) Ignore region $IR = R(x_g^{'}, y_g^{'}, \sigma_2 w^{'}, \sigma_2 h^{'}) \setminus CR$, that is, the box region with the CR area removed. The pixels in IR are marked as ignored and do not participate in training, as indicated by the yellow area in Figure 3.

(3) Outer region OR, which is the part of the entire feature map excluding CR and IR. The pixels in OR are designated as negative samples, as indicated by the gray area in Figure 3.

Densebox [21] introduced the "gray area" of balanced sampling, which helps us train anchor position prediction branch N_L, but it only applies to a single feature map. Because we use multiple feature levels of a FPN, we also consider the impact of adjacent feature maps. Specifically, each level of feature map should only target objects at a specific scale range. Therefore, we only assign a CR region in the feature map when the feature map matches the scale range of the target object. The same areas in adjacent levels are set to IR,

as shown in Figure 3. When multiple objects overlap, CR can inhibit IR, and IR can inhibit OR. Because CR usually occupies a small part of the entire feature map, we use focal loss [27] to train the position branch.

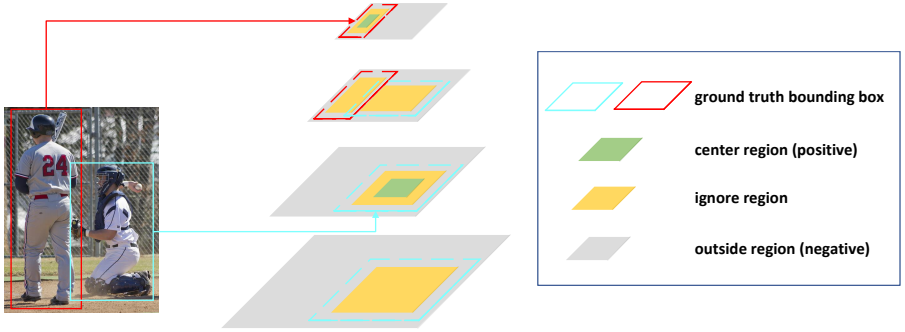

Fig. 3 Multi-level anchor positioning. The ground truth is assigned to different feature levels according to the scale

To train the anchor shape branch, first we need to match the anchors to the ground truth. Next, we predict the width and height of the anchor so that it can best cover the matched ground truth. We define the IOU between the variable anchor $a_{wh} = (x_0, y_0, w, h)|w > 0, h > 0$ and ground truth $gt = (x_g, y_g, w_g, h_g)$ as vIOU, as follows:

$$vIOU(a_{wh}, gt) = \max_{w>0,h>0} IOU_{normal}(a_{wh}, gt) \tag{5}$$

In this equation, IOU_{normal} is used to calculate the intersection of union (IOU) between the variable anchor and the ground truth, and both w and h are variables. Because a_{wh} is variable and the calculation of $vIOU(a_{wh}, gt)$ is relatively complicated, it is difficult to realize it in an end-to-end network. Therefore, we use an enumeration method to approximate it. Given (x_0, y_0), we sample some common values of w and h to simulate all enumerations of w and h. Then, we use gt to calculate the IOU of these sampled anchors, and use the maximum value as an approximation of $vIOU(a_{wh}, gt)$. In our experiment, we sampled nine groups of (w, h) samples to estimate vIOU during training. Specifically, we adopted nine groups of different sizes and aspect ratios used by RetinaNet [27]. In theory, the more groups sampled, the more accurate the approximation. However, the computational cost also increases. We use a variant of the bounded IOU loss [43] to optimize the shape prediction branch because the anchor position is fixed, and hence only w and h are optimized instead of all w, y, w, h. The loss is:

$$L_{shape} = L_1(1 - min(\frac{w}{w_g}, \frac{w_g}{w})) + L_1(1 - min(\frac{h}{h_g}, \frac{h_g}{h})) \tag{6}$$

where (w, h) and (w_g, h_g) represent the predicted anchor shape and the corresponding ground truth shape, and L_1 is smooth L_1 loss.

3.2 SAC

To better detect multi-scale pedestrians at the micro level, the proposed method uses SAC to replace standard convolution in the backbone network. SAC uses convolution kernels with different atrous rates to process the same input features and uses a switch function to collect the results. In this way, our aim is to use an atrous convolution with an atrous rate of 1 for smaller-scale pedestrian target positions, and an atrous convolution with a large atrous rate for larger-scale pedestrian target location. Figure 4 shows the SAC structure, which consists of three main components: the SAC component itself and two global context modules, one before and after it.

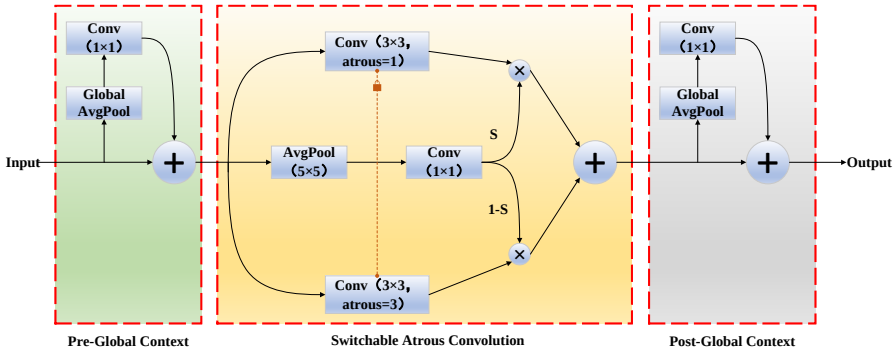

Fig. 4 Switchable Atrous Convolution(SAC)

We first introduce the SAC component. Let $\boldsymbol{y} = \boldsymbol{Conv}(\boldsymbol{x}, \boldsymbol{w}, r)$ denote a convolution operation with $\boldsymbol{x}$ as input, $\boldsymbol{y}$ as output, r as the atrous rate, and weight $\boldsymbol{w}$. Then, we can convert the convolutional layer to SAC, as follows.

$$\begin{aligned} \boldsymbol{Conv}(\boldsymbol{x}, \boldsymbol{w}, 1) \overset{Convert}{\underset{toSAC}{\longrightarrow}} & \boldsymbol{S}(\boldsymbol{x}) \cdot \boldsymbol{Conv}(\boldsymbol{x}, \boldsymbol{w}, 1) \\ & + (1 - \boldsymbol{S}(\boldsymbol{x})) \cdot \boldsymbol{Conv}(\boldsymbol{x}, \boldsymbol{w} + \Delta\boldsymbol{w}, r) \end{aligned} \tag{7}$$

In this equation, r is the hyperparameter of SAC, $\Delta\boldsymbol{w}$ is a trainable weight, and the switch function $\boldsymbol{S}(\cdot)$ is implemented by a 5×5 average pooling layer and a 1×1 convolutional layer, as shown in Figure 4. After the switch function is obtained, according to formula (7), it is multiplied and added by the output results of the atrous convolution with atrous rate 1 and r respectively. The results are collected so that the model can use atrous convolutions with different atrous rates depending on pedestrians at different scales. The switch function depends on the input and location, and hence the backbone can adapt to different ranges as needed. We set $r = 3$ in the experiment. Simultaneously, we use a locking mechanism to set the weight when the atrous ratio is 1 to $\boldsymbol{w}$, and the other weight is set to $\boldsymbol{w} + \Delta\boldsymbol{w}$. This is because the detector usually uses pre-training to initialize the weights. However, when the SAC layer

is converted from the standard convolution layer, it lacks the weights of the larger atrous rate. Because pedestrian targets with different proportions can be roughly detected with the same weight and different atrous rates, the weights in the pre-training model can be used to initialize the missing weights. We use $\boldsymbol{w}+\Delta\boldsymbol{w}$ to represent the missing weight, where w comes from the pre-training weight and $\Delta\boldsymbol{w}$ is initialized to 0.

As shown in Figure 4, we insert a global context module before and after the SAC component, and our aim is to apply the global information before and after the switch function. This module passes the input through a global average pooling and 1×1 convolution, and then adds the result back to the main stream. We use deformable convolution [10] to replace the two convolution operations in equation (7), and their offset functions are not shared. When loaded from the pre-training backbone, these functions are initialized to 0. We use SAC on ResNet and ResNeXt [50] by replacing all 3×3 convolutional layers in the backbone. The weight and offset in the global context module are initialized to 0, the weight in the switch function $\boldsymbol{S}$ is initialized to 0, the bias is set to 1, and $\Delta\boldsymbol{w}$ is initialized to 0.

4 Experimental results and analysis

4.1 Datasets

The experiments in this paper use the COCOPersons dataset, Caltech, and CityPersons pedestrian dataset for training and verification.

The COCOPersons dataset is a subset of the MS COCO dataset, and its images only include the ground truth of people. The other 79 categories were ignored in the evaluation. The dataset was split into 64,115 images for the training set (which includes 257,252 pedestrian targets; each image contains 4.01 pedestrian targets on average), and 2,693 images for the validation set.

The Caltech pedestrian dataset consists of approximately 10 hours of 640x480 30Hz video taken from a vehicle driving through regular traffic in an urban environment. About 250,000 frames (in 137 approximately minute long segments) with a total of 350,000 bounding boxes and 2300 unique pedestrians were annotated.

The CityPersons dataset is a subset of Cityscapes which only consists of person annotations. There are 2975 images for training, 500 and 1575 images for validation and testing. The density of pedestrians in the dataset is very high, and the average number of pedestrians in an image is 7. What's more, scenarios of the datasets are rich, and it contains multiple occlusion cases.

4.2 Evaluation metrics

The evaluation metrics in the COCOPersons dataset use the average precision (AP) of the MS COCO dataset as an indicator. Specifically, this study uses

mAP (mean average precision), AP_S (AP for small pedestrian targets: $area < 32^2$), AP_M (AP for medium pedestrian targets: $32^2 < area < 96^2$), and AP_L (AP for large pedestrian targets: $area > 96^2$). We use the AR value as an indicator when verifying the recall of GARPN and RPN. AR is the average value of recalls under different IOU thresholds (from 0.5 to 0.95). The AR of 100 proposals per image is represented as AR_{100}. We calculated the AR of small, medium, and large targets (AR_S, AR_M, AR_L) for 100 proposals.

The experiments in this paper on the Caltech pedestrian dataset are based on the Caltech evaluation criterion: the average false positive per image (FPPI) pedestrian missed detection rate between $\left[10^{-2}, 10^{0}\right]$, denoted by MR^{-2}. According to the data division standard of Caltech test set, this paper mainly selects: Near, Medium and Far subsets represent the test subsets whose pedestrian height ranges are greater than 80 pixels, between 30 and 80 pixels, and between 20 and 30 pixels, respectively.

The experiments in this paper on the CityPersons pedestrian dataset are based on the CityPersons evaluation criterion: the average false positive per image (FPPI) pedestrian missed detection rate between $\left[10^{-2}, 10^{0}\right]$, denoted by MR^{-2}. According to the size of the area occupied by the target, the CityPersons dataset is divided into three subsets, Small, Medium and Large, with the pixel area 32^2 and 96^2 as the boundary, respectively, to verify the detection performance of the algorithm for small, medium and large scale pedestrians.

4.3 Implementation details

The experiment was conducted using PyTorch [5], CUDA 10.1, and the MMDetection 2.0 target detection library, with ResNet50 as the pre-training weights. The training was conducted on two NVIDIA RTX 2080 Ti GPUs. Training was performed over 12 epochs on the COCOPersons dataset, Caltech,and CityPersons pedestrian dataset. Using the stochastic gradient descent method, the initial learning rate was set to 0.0025, and after the 8^{th} and 11^{th} epochs, it was reduced by one-tenth. The input image size was 1333 $\times$ 800 pixels. We set $\sigma_1 = 0.2$ and $\sigma_2 = 0.5$ in GARPN. In the multi-task loss function, we set $\lambda_1 = 1$ and $\lambda_2 = 0.1$ to balance the position and shape prediction branches. We uniformly set the momentum factor to 0.9 and the weight attenuation factor to 0.0001 to prevent the model from overfitting.

4.4 Experimental results and analysis

4.4.1 Testing on the COCOPersons dataset

Table 2 compares the evaluation results of the proposed model with those of other detection methods on the COCOPersons dataset. Using Faster R-CNN as the baseline, after embedding the method proposed in this paper, the accuracy of pedestrian detection is improved by 3.7 AP under the same conditions.

Moreover, the detection accuracy of pedestrian targets at large, medium, and small scales is improved by 1.7 AP, 2.5 AP, and 6.8 AP, respectively. These results also verifies that the proposed model substantially improves the accuracy of multi-scale pedestrian detection. Moreover, Table 2 reveals that the detection results of this model are better than those of other current detection methods. We then replaced the backbone with ResNeXt-50 and evaluated the results again, obtaining a pedestrian detection accuracy of 57.7 AP. Moreover, the pedestrian detection accuracies at the three scales were 37.2 AP, 63.8 AP, and 77.9 AP, respectively. We present a visual result of multi-scale pedestrian

Table 2 Comparison of the results on the COCOPersons dataset

Method	Backbone	mAP	AP_S	AP_M	AP_L
RetinaNet [27]	ResNet-50	53.2	34.2	60.6	72.4
Mask R-CNN [18]	ResNet-50	55.3	37.0	62.8	72.8
Cascade R-CNN [4]	ResNet-50	56.6	36.9	63.6	74.3
Faster R-CNN [38]	ResNet-50	53.6	35.3	61.2	70.9
SINPER [39]	ResNet-50	54.8	35.9	62.7	73.5
SAFNet [23]	ResNet-50	55.0	36.0	62.8	73.9
YOLOF [7]	ResNet-50	55.8	36.3	62.9	74.9
Dynamic R-CNN [54]	ResNet-50	56.3	36.5	63.4	75.4
CBNet [32]	ResNeXt-50	56.8	36.9	63.7	75.5
SCNet [44]	ResNet-50	56.5	36.8	63.7	75.2
Sparse R-CNN [41]	ResNet-50	57.0	37.0	63.9	77.5
Ours	**ResNet-50**	**57.3**	**37.0**	**63.7**	**77.7**
Ours	**ResNeXt-50**	**57.7**	**37.2**	**63.8**	**77.9**

detection on the COCOPersons dataset in Figure 5. As shown in this figure, pedestrian targets at different scales are accurately detected; in particular, small pedestrian targets in the distance are also accurately located.

We also compared the results of the model proposed in this paper with other methods for solving the problem of pedestrian scale differences on the COCOPersons dataset. As presented in Table 3, the pedestrian detection accuracy of the model in this paper is much higher than those of the balanced feature pyramid (BFP) and the recursive feature pyramid (RFP) at large, medium, and small scales. Compared with the multi-scale feature enhancement modules the augmentation feature pyramid networks (AugFPN) and the feature-aligned pyramid network (FaPN), the proposed model has a comparable small-scale pedestrian detection accuracy, whereas its medium- and large-scale pedestrian detection accuracies are substantially improved. These results also demonstrate that the model proposed in this paper better solves the problem of pedestrian scale differences than other multi-scale methods.

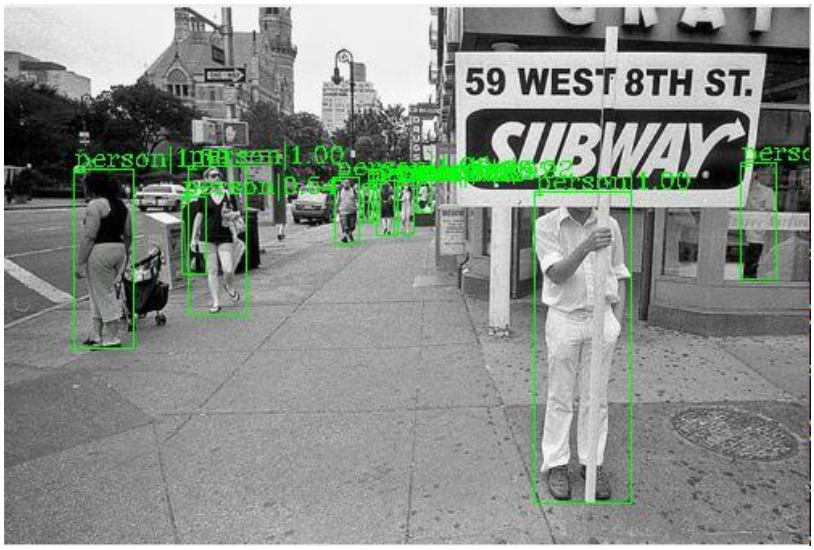

Fig. 5 Visual results on the COCOPersons dataset

Table 3 Comparison with other multi-scale methods on the COCOPersons dataset

Method	Backbone	mAP	AP_S	AP_M	AP_L
RetinaNet+NasFPN [16]	ResNet-50	53.8	36.0	60.7	71.5
Faster R-CNN+BFP [36]	ResNet-50	54.6	36.8	61.4	71.9
Faster R-CNN+RFP [37]	ResNet-50	55.7	36.5	62.7	74.7
Faster R-CNN+MFBE [20]	ResNet-50	54.7	37.1	61.7	71.7
Faster R-CNN+AugFPN [17]	ResNet-50	55.0	37.0	62.1	73.5
Faster R-CNN+FaPN [22]	ResNet-50	56.5	37.3	63.5	75.6
Ours	**ResNet-50**	**57.3**	**37.0**	**63.7**	**77.7**

4.4.2 Testing on the Caltech pedestrian dataset

In order to laterally compare the effectiveness of the proposed model for multi-scale pedestrian detection, an experimental comparison is conducted on the Caltech dataset with the current pedestrian detection methods that perform well. As shown in Table 4, the MR^{-2} of this model on the Near, Medium and Far subsets are 0.45%, 13.78%, and 48.85%, respectively. Compared with Faster R-CNN+ATT, the of our model on the Near, Medium and Far subsets are reduced by 0.98%, 26.97%, and 42.09%, respectively. Compared with the current state-of-the-art TLL-TFA method, the MR^{-2} are reduced by 0.27%, 9.14% and 55.24% on the subsets of Near, Medium and Far, respectively. This also shows that the proposed model has better detection effect for pedestrian objects of different scales.

Table 4 Comparison of the results on the Caltech Pedestrain dataset

Method	Near	Medium	Far
Faster R-CNN+ATT [57]	1.43	40.75	90.94
RPN+BF [55]	2.26	53.93	100
AR-Ped [2]	1.37	49.31	100
TLL-TFA [40]	0.72	22.92	60.09
Ours	**0.45**	**13.78**	**48.85**

4.4.3 Testing on the CityPersons pedestrian dataset

At the same time, we compared the proposed method with other multi-scale pedestrian detection methods on CityPersons dataset. As shown in Table 5, the MR^{-2} of this model on the Small, Medium and Large subsets are 12.1%, 2.6%, and 5.5%, respectively. Compared with Faster R-CNN, the MR^{-2} of our model on the Small, Medium and Large subsets are reduced by 13.5%, 4.6%, and 2.4%, respectively. Compared with MagnifierNet, the MR^{-2} of our model on the Small, Medium and Large subsets are reduced by 0.5%, 2.9%, and 2.2%, respectively. Compared with DHRNet with excellent performance, the MR^{-2} are reduced by 1.3%, 0.1% and 0.7% on the subsets of Small, Medium and Large, respectively. This shows that the model proposed in this paper also has excellent detection effect for pedestrian targets of different scales on the CityPersons dataset.

Table 5 Comparison of the results on the CityPersons Pedestrain dataset

Method	Backbone	Small	Medium	Large
Fater R-CNN [56]	VGG-16	25.6	7.2	7.9
CSP [30]	ResNet-50	16	3.7	6.5
MagnifierNet [8]	ResNet-101	12.6	5.5	7.7
PRF-Ped [42]	ResNet-50	12.9	3.9	5.8
DHRNet [12]	DHRNet-W18	13.4	2.7	6.2
Ours	**ResNet-50**	**12.1**	**2.6**	**5.5**

4.4.4 Ablation experiments

To explore the impact of the three modules on the model detection performance, we conducted ablation experiments on the COCOPersons dataset, and the results are presented in Table 6. At the macro level, we first introduced GARPN separately into a Faster R-CNN based on FPN, and the accuracy of pedestrian detection at three scales was improved by 0.9 AP, 2.0 AP, and 2.2 AP, respectively. At the same time, there is only a small increase in the amount

of model size. The results verify that, compared with the traditional RPN fixed anchor generation scheme, predicting the position and shape of the anchor through the feature map can better adapt to the common scale-difference problem in pedestrian detection. Then, at the micro level, we replaced the 3×3 convolution in the backbone with SAC. Although the model size increases by 19.2 MB, the improvement in detection accuracy is also significant. Pedestrian detection accuracy is improved at the large, medium, and small scales. In particular, the accuracy of large-scale pedestrian detection is improved by 4.3 AP. This result shows that SAC sets up the atrous convolution with different atrous rates, and then allocates the weight of the results obtained by the atrous convolution with different atrous rates using the switch function according to the pedestrian targets with different scales, and collects the results. In this way, smaller scale pedestrian targets will use more the results of atrous convolution with an atrous ratio of 1, and larger scale pedestrian targets will use more the results of atrous convolution with a larger atrous ratio. So as to get better multi-scale pedestrian detection effect. Finally, we introduced the Double Head to the proposed method, which improves the pedestrian detection accuracy by 0.3 AP, verifying that the Double Head, that is, using an fc-head to focus on classification tasks and a conv-head to focus on regression tasks, is beneficial for detection.

Table 6 Ablation experiment results of each module

GARPN	SAC	Double Head	mAP	AP_S	AP_M	AP_L	Model size/MB
			53.6	35.3	61.2	70.9	333.4
√			55.3	36.2	63.2	73.1	334.9
√	√		57.0	36.6	63.5	77.4	354.1
√	√	√	57.3	37.0	63.7	77.7	378.5

We compare the recall results on the COCOPersons dataset in Table 7. Compared with the traditional anchoring scheme RPN, the GARPN of the model in this paper increases the AR_{100} by 3.4, and increases the recall rates of pedestrian targets at large, medium, and small scales by 4.3, 3.0, and 2.6 respectively. This verifies that GARPN predicts the possible locations of anchors using the feature map, which greatly reduces the possibility of anchors appearing in non-interest areas and background areas. We show the effect of predicting anchors and the visualization of the output of the two branches. As shown in Figure 6, the anchors are more focused on the pedestrian targets in the image, which also provides a basis for obtaining the subsequent proposals. Meanwhile, compared with other related methods, our method has better performance.

We have also explored the impact of each component of the SAC module more deeply, as detailed in Table 8. First, we introduced the SAC separately, and the pedestrian detection accuracy was 56.5 AP. On this basis, we removed the deformable convolution (SAC-DCN), and the detection accuracy was re-

Table 7 Recall results on COCOPersons dataset

Method	Backbone	AR_{100}	AR_S	AR_M	AR_L
RPN	ResNet-50	62.1	46.1	68.8	78.3
AEMS-RPN [45]	ResNet-50	62.7	46.4	69.7	78.4
Attention-RPN [15]	ResNet-50	63.0	48.2	70.5	78.8
RPN Prototype Alignment [58]	ResNet-50	63.6	48.7	70.9	79.5
GARPN	ResNet-50	65.5	50.4	71.8	80.9

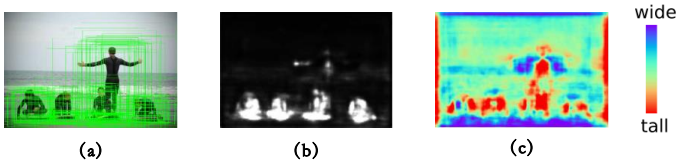

Fig. 6 Anchor prediction effect. (a) Input image and predicted anchor effect, (b) predicted anchor position probability map, and (c) predicted anchor aspect ratio

duced by 1.0 AP. Then, we removed the global context module (SAC-DCN-global), and the detection accuracy was reduced by 1.3 AP. This shows that adding global context information before the SAC component has a positive impact on detection performance. This is because global information enables the switch function to make more stable switching predictions. Finally, we use $\boldsymbol{\Delta w}$ only instead of $\boldsymbol{w} + \boldsymbol{\Delta w}$ for the weight of the convolution with an atrous rate of 3. The importance of the locking mechanism (SAC-DCN-locking) is verified because after the locking mechanism was removed, the detection accuracy dropped by 0.9 AP. Figure 7 shows a visualization of the output of the last switch function, where the darker area indicates that the switch function collects more results from atrous convolution with a larger atrous rate. Comparing this result with the original image, we can observe that the output result of the switch function is highly aligned with the original image. That is, for larger-scale pedestrian targets, atrous convolution with a larger atrous rate is used. On the contrary, for a smaller-scale pedestrian target, atrous convolution with a smaller atrous rate is used.

Table 8 Ablation experiment results for each component of the SAC module

Method	Backbone	mAP	AP_S	AP_M	AP_L
SAC	ResNet-50	56.5	36.1	63.0	77.0
SAC-DCN	ResNet-50	55.5	35.8	62.1	75.2
SAC-DCN-global	ResNet-50	54.2	35.4	61.4	74.2
SAC-DCN-locking	ResNet-50	54.6	35.7	62.0	73.6

Fig. 7 Visualization of the outputs of the switch functions in SAC

5 Conclusions

In view of the widespread problem of scale differences in pedestrian detection, this paper systematically summarized the current mainstream detectors that still have shortcomings at the macro and micro levels. We then proposed the multi-scale pedestrian detection model, which considers both the macro and micro levels. At the macro level, predicting the location and shape of the anchor through the feature map enables the model to better adapt to pedestrian targets at different scales. At the micro level, SAC is used to replace standard convolution in the backbone, and the atrous rate of atrous convolution is adjusted according to the target location of pedestrians at different scales. Finally, by reasonably assigning classification and regression tasks, the pedestrian detection performance of the model is further improved.

6 Acknowledgements

This work is supported by the National Natural Science Foundation of China (61872042, 61972375, 62172045), the Science and Technology Project of Beijing Municipal Commission of Education (KM202111417009, KM201811417005), the Major Project of Technological Innovation 2030 - "New Generation Artificial Intelligence" (2018AAA0100800), Premium Funding Project for Academic Human Resources Development in Beijing Union University(BPHR2020AZ01, BPH2020EZ01), the Key Project of Beijing Municipal Commission of Education (KZ201911417048).

7 Data availability statement

The data that support the findings of this study are available from the corresponding author, Ning He, upon reasonable request.

References

1. Alsaih, K., Yusoff, M.Z., Tang, T.B., Faye, I., Mériaudeau, F.: Performance evaluation of convolutions and atrous convolutions in deep networks for retinal disease segmentation on optical coherence tomography volumes. In: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), pp. 1863–1866. IEEE (2020)
2. Brazil, G., Liu, X.: Pedestrian detection with autoregressive network phases. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7231–7240 (2019)
3. Cai, J., Lee, F., Yang, S., Lin, C., Chen, H., Kotani, K., Chen, Q.: Pedestrian as points: An improved anchor-free method for center-based pedestrian detection. IEEE Access **8**, 179666–179677 (2020)
4. Cai, Z., Vasconcelos, N.: Cascade r-cnn: Delving into high quality object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6154–6162 (2018)
5. Chen, K.M., Cofer, E.M., Zhou, J., Troyanskaya, O.G.: Selene: a pytorch-based deep learning library for sequence data. Nature methods **16**(4), 315–318 (2019)
6. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
7. Chen, Q., Wang, Y., Yang, T., Zhang, X., Cheng, J., Sun, J.: You only look one-level feature. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 13039–13048 (2021)
8. Cheng, Q., Chen, M., Wu, Y., Chen, F., Lin, S.: Magnifiernet: Learning efficient small-scale pedestrian detector towards multiple dense regions. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 1483–1490. IEEE (2021)
9. Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. Advances in neural information processing systems **29** (2016)
10. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: Proceedings of the IEEE international conference on computer vision, pp. 764–773 (2017)
11. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol. 1, pp. 886–893. Ieee (2005)
12. Ding, M., Zhang, S., Yang, J.: Learning a dynamic high-resolution network for multi-scale pedestrian detection. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 9076–9082. IEEE (2021)
13. Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: An evaluation of the state of the art. IEEE transactions on pattern analysis and machine intelligence **34**(4), 743–761 (2011)
14. Du, X., El-Khamy, M., Lee, J., Davis, L.: Fused dnn: A deep neural network fusion approach to fast and robust pedestrian detection. In: 2017 IEEE winter conference on applications of computer vision (WACV), pp. 953–961. IEEE (2017)
15. Fan, Q., Zhuo, W., Tang, C.K., Tai, Y.W.: Few-shot object detection with attention-rpn and multi-relation detector. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4013–4022 (2020)
16. Ghiasi, G., Lin, T.Y., Le, Q.V.: Nas-fpn: Learning scalable feature pyramid architecture for object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 7036–7045 (2019)
17. Guo, C., Fan, B., Zhang, Q., Xiang, S., Pan, C.: Augfpn: Improving multi-scale feature learning for object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 12595–12604 (2020)
18. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision, pp. 2961–2969 (2017)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778 (2016)

20. He, Y., He, N., Zhang, R., Yan, K., Yu, H.: Multi-scale feature balance enhancement network for pedestrian detection. Multimedia Systems **28**(3), 1135–1145 (2022)
21. Huang, L., Yang, Y., Deng, Y., Yu, Y.: Densebox: Unifying landmark localization with end to end object detection. arXiv preprint arXiv:1509.04874 (2015)
22. Huang, S., Lu, Z., Cheng, R., He, C.: Fapn: Feature-aligned pyramid network for dense image prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 864–873 (2021)
23. Jin, Z., Liu, B., Chu, Q., Yu, N.: Safnet: A semi-anchor-free network with enhanced feature pyramid for object detection. IEEE Transactions on Image Processing **29**, 9445–9457 (2020)
24. Kim, M., Ilyas, N., Kim, K.: Amsaseg: An attention-based multi-scale atrous convolutional neural network for real-time object segmentation from 3d point cloud. IEEE Access **9**, 70789–70796 (2021)
25. Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., Sun, J.: Detnet: A backbone network for object detection. arXiv preprint arXiv:1804.06215 (2018)
26. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117–2125 (2017)
27. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988 (2017)
28. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision, pp. 740–755. Springer (2014)
29. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision, pp. 21–37. Springer (2016)
30. Liu, W., Liao, S., Ren, W., Hu, W., Yu, Y.: High-level semantic feature detection: A new perspective for pedestrian detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 5187–5196 (2019)
31. Liu, X., Chen, H.X., Liu, B.Y.: Dynamic anchor: A feature-guided anchor strategy for object detection. Applied Sciences **12**(10), 4897 (2022)
32. Liu, Y., Wang, Y., Wang, S., Liang, T., Zhao, Q., Tang, Z., Ling, H.: Cbnet: A novel composite backbone network architecture for object detection. In: Proceedings of the AAAI conference on artificial intelligence, vol. 34, pp. 11653–11660 (2020)
33. Ma, W., Tian, T., Xu, H., Huang, Y., Li, Z.: Aabo: Adaptive anchor box optimization for object detection via bayesian sub-sampling. In: European Conference on Computer Vision, pp. 560–575. Springer (2020)
34. Ming, Q., Zhou, Z., Miao, L., Zhang, H., Li, L.: Dynamic anchor learning for arbitrary-oriented object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 2355–2363 (2021)
35. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern recognition **29**(1), 51–59 (1996)
36. Pang, J., Chen, K., Shi, J., Feng, H., Ouyang, W., Lin, D.: Libra r-cnn: Towards balanced learning for object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 821–830 (2019)
37. Qiao, S., Chen, L.C., Yuille, A.: Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10213–10224 (2021)
38. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
39. Singh, B., Najibi, M., Davis, L.S.: Sniper: Efficient multi-scale training. Advances in neural information processing systems **31** (2018)
40. Song, T., Sun, L., Xie, D., Sun, H., Pu, S.: Small-scale pedestrian detection based on topological line localization and temporal feature aggregation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 536–551 (2018)

41. Sun, P., Zhang, R., Jiang, Y., Kong, T., Xu, C., Zhan, W., Tomizuka, M., Li, L., Yuan, Z., Wang, C., et al.: Sparse r-cnn: End-to-end object detection with learnable proposals. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 14454–14463 (2021)
42. Tan, Y., Yao, H., Li, H., Lu, X., Xie, H.: Prf-ped: Multi-scale pedestrian detector with prior-based receptive field. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 6059–6064. IEEE (2021)
43. Tychsen-Smith, L., Petersson, L.: Improving object localization with fitness nms and bounded iou loss. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6877–6885 (2018)
44. Vu, T., Kang, H., Yoo, C.D.: Scnet: Training inference sample consistency for instance segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 2701–2709 (2021)
45. Wang, H., Li, Y., Wang, S.: Fast pedestrian detection with attention-enhanced multi-scale rpn and soft-cascaded decision trees. IEEE Transactions on Intelligent Transportation Systems **21**(12), 5086–5093 (2019)
46. Wang, J., Chen, K., Yang, S., Loy, C.C., Lin, D.: Region proposal by guided anchoring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2965–2974 (2019)
47. Wang, W.: Adapted center and scale prediction: more stable and more accurate. arXiv preprint arXiv:2002.09053 (2020)
48. Wang, Z., Ji, S.: Smoothed dilated convolutions for improved dense prediction. Data Mining and Knowledge Discovery **35**(4), 1470–1496 (2021)
49. Wu, Y., Chen, Y., Yuan, L., Liu, Z., Wang, L., Li, H., Fu, Y.: Rethinking classification and localization for object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10186–10195 (2020)
50. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1492–1500 (2017)
51. Yang, T., Zhang, X., Li, Z., Zhang, W., Sun, J.: Metaanchor: Learning to detect objects with customized anchors. Advances in neural information processing systems **31** (2018)
52. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122 (2015)
53. Yu, F., Koltun, V., Funkhouser, T.: Dilated residual networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 472–480 (2017)
54. Zhang, H., Chang, H., Ma, B., Wang, N., Chen, X.: Dynamic r-cnn: Towards high quality object detection via dynamic training. In: European conference on computer vision, pp. 260–275. Springer (2020)
55. Zhang, L., Lin, L., Liang, X., He, K.: Is faster r-cnn doing well for pedestrian detection? In: European conference on computer vision, pp. 443–457. Springer (2016)
56. Zhang, S., Benenson, R., Schiele, B.: Citypersons: A diverse dataset for pedestrian detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3213–3221 (2017)
57. Zhang, S., Yang, J., Schiele, B.: Occluded pedestrian detection through guided attention in cnns. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 6995–7003 (2018)
58. Zhang, Y., Wang, Z., Mao, Y.: Rpn prototype alignment for domain adaptive object detector. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12425–12434 (2021)
59. Zhong, Y., Wang, J., Peng, J., Zhang, L.: Anchor box optimization for object detection. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1286–1294 (2020)
60. Zhu, Y., Wang, J., Zhao, C., Guo, H., Lu, H.: Scale-adaptive deconvolutional regression network for pedestrian detection. In: Asian Conference on Computer Vision, pp. 416–430. Springer (2017)