A SINGLE-STAGE PEDESTRIAN DETECTOR BASED ON SSD WITH MULTI-SCALE FEATURE EXTRACTION AND RESIDUAL BLOCK

HOANH NGUYEN

Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

E-mail: nguyenhoanh@iuh.edu.vn

ABSTRACT

Pedestrian detection is a key problem in many intelligent transport systems. In driving environment, apart from the detection accuracy, the inference speed is also a large concern. Although popular two-stage object detectors such as Faster R-CNN have achieved significant improvements in pedestrian detection accuracy, it is still slow for real-time pedestrian detection in driving environment. On the other hand, popular one-stage object detectors such as SSD have not achieved competitive detection accuracy on pedestrian detection benchmarks. This paper proposes a one-stage detector for real-time pedestrian detection in driving environment. The proposed approach is based on popular SSD framework. To improve the detection accuracy, the backbone network in original SSD framework is replaced by the backbone sub-network based on DenseNets structure, which includes stem module, dense blocks, and transition layers. With dense connection in DenseNet architecture, the proposed approach can achieve higher accuracy with fewer parameters compared with ResNet architecture. In the detection sub-network, enhanced feature extraction subnet takes convolution layers generated by the backbone sub-network to generate enhanced feature maps by fusing operation, atrous convolution and deconvolution operation. Enhanced feature maps can enhance the detection performance of multi-scale pedestrian detection. In addition, residual blocks are added before each prediction layer to reduce the computational cost and improve the detection accuracy. Experimental results on Caltech and CityPersons dataset show that the proposed approach achieves better accuracy compared with popular two-stage detectors while being faster.

Keywords: *Single-Stage Detector, Two-Stage Detector, Pedestrian Detection, Deep Learning, Feature Extraction*

1. INTRODUCTION

Pedestrian detection is a key problem in many intelligent transportation systems such as autonomous driving systems, video surveillance systems, and so on. A pedestrian detection framework usually consists of three steps: Proposal generation, feature extraction, and classification. Proposal generation step takes the input image and generates regions where pedestrians may exist. Feature extraction step computes the features for each proposed region. Classification step further classifies proposed region into pedestrian and background class. Traditional pedestrian detection approaches are usually based on hand-crafted features to locate pedestrian in an image. Since AlexNet [32] made a significant improvement on ImageNet dataset, various deep learning-based approaches represented by convolutional neural networks (CNNs) have been widely applied in many visual tasks, including pedestrian detection. As an object detection algorithm based on deep learning, R-CNN firstly introduced CNN into object detection. Following R-CNN, Faster-RCNN proposed region proposal network to generate proposals in a unified framework. With the success of Faster R-CNN on general object detection, numerous adapted Faster-RCNN detectors were proposed and demonstrated better accuracy for pedestrian detection. However, when the inference speed is concerned, Faster-RCNN is still unsatisfactory because it requires two-stage processing, including proposal generation and proposal classification. As a representative one-stage object detector, Single Shot MultiBox Detector (SSD) discards the second stage of Faster-RCNN and directly regresses the default anchors into detection boxes. Although SSD framework is faster than Faster R-CNN framework, SSD has not presented competitive results on pedestrian detection benchmarks.

Motivated by the above research problems, this paper proposes a single-stage pedestrian detection framework based on SSD. There are two sub-networks in the proposed framework, including the backbone sub-network and the detection sub-network. In the backbone sub-network, the base convolution layers are generated by the improved architecture based on DenseNets structure. With dense connection as in original DenseNets architecture, the proposed approach can achieve higher accuracy with fewer parameters compared with ResNet architecture. In the detection sub-network, enhanced feature extraction subnet takes convolution layers generated by the backbone sub-network to generate enhanced feature maps by fusing operation, atrous convolution and deconvolution operation. Enhanced feature extraction subnet takes full advantage of the relationship between different feature layers of the backbone sub-network, thus enhancing detection performance of multi-scale pedestrian detection. Furthermore, residual blocks are added before each prediction layer. Residual blocks can reduce the computational cost while the detection accuracy is improved. Experimental results on Caltech and CityPersons dataset show that the proposed approach achieves better accuracy compared with popular two-stage detectors on pedestrian detection while being faster.

This paper is organized as follows: an overview of previous methods is presented in Section 2. Section 3 describes detail the proposed method. Section 4 demonstrates experimental results. Finally, the conclusion is made in Section 5.

2. RELATED WORK

2.1 Deep Learning Object Detection Approach

Deep learning-based object detection approaches can be divided into two groups: one-stage approaches and two-stage approaches. Two-stage approaches first generate proposals and then classify these proposals into object and background class. Popular two-stage approaches include R-CNN [19], Fast RCNN [20], Faster R-CNN [21] and R-FCN [22]. R-CNN framework uses selective search [23] to first generate potential proposals in an image and then perform classification on the proposed regions. While the R-CNN framework surpassed previous detectors by a large margin, its speed is limited by the need for object proposal generation and repeated convolutional neural network (CNN) evaluation. Fast-RCNN framework introduced the ideas of back-propagation through the ROI pooling layer and multi-task learning of a classifier and a bounding box regressor. However, it still depends on bottom-up proposal generation. More recently, the Faster-RCNN framework has addressed the generation of object proposals and classifier within a single neural network, leading to a significant speedup for proposal detection. R-FCN framework further improves speed and accuracy by removing fully-connected layers and adopting position-sensitive score maps for final detection. One-stage object detection approaches eliminate the proposal generation process for real-time detection. Popular one-stage approaches include YOLO [24] and SSD [6]. YOLO framework uses a single feed-forward convolutional network to directly predict object classes and locations. Comparing with two-stage approaches, YOLO no longer requires a second per-region classification operation so that it is extremely fast. SSD framework improves YOLO in several aspects, including using small convolutional filters to predict categories and anchor offsets for bounding box locations, using pyramid features for prediction at different scales, and using default boxes and aspect ratios for adjusting varying object shapes. In this paper, the proposed approach is built upon the SSD framework and thus it inherits the speed and accuracy advantages of SSD, while produces smaller and more flexible models.

2.2 Pedestrian Detection

Traditional pedestrian detection approaches are usually based on hand-crafted features to locate pedestrian in an image. The authors in [25] proposed an efficient feature transform that removes correlations in local neighborhoods. Li et al. [26] proposed to construct the co-occurrence of multiple channel features in local image neighborhoods for pedestrian detection. More specific, a binary pattern of feature co-occurrence is represented by combining the binary variables quantized from each channel feature, and the spatial information is incorporated by selecting the neighbors to jointly represent the feature co-occurrence in a local image block. Although traditional pedestrian detection approaches can achieve high performance, current leading pedestrian detection approaches are based on deep learning. In [27], the authors proposed the DeepParts framework, which is able to handle low IoU positive proposals that shift away from ground

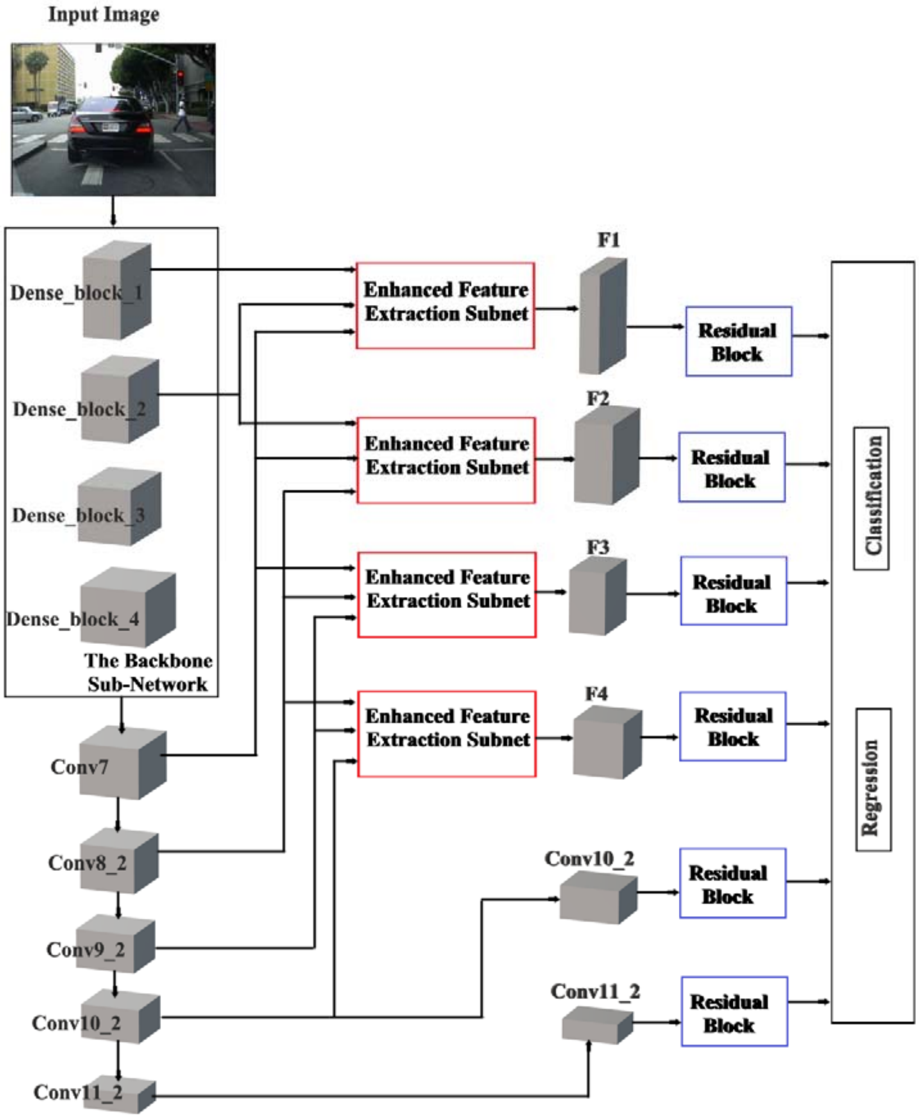

Figure 1: The Overall Architecture of The Proposed Approach.

truth and can detect pedestrian by observing only a part of a proposal. Zhou et al. [28] introduced an efficient network in which the part detectors are learned jointly to exploit part correlations. Zhang et al. [29] proposed a simple and compact method based on the Faster-RCNN architecture for occluded pedestrian detection. The author in [30] proposed to use deconvolutional modules on the base convolution layers to bring additional context information which is more effective to detect small-scale pedestrians. In the region of interest pooling process, different feature maps at different scales are used to produce high quality region proposals. Wang et al. [31] design a repulsion loss for pedestrian detection in crowd scenes to constrain the predicted boxes close to the ground-truth while being away from the other boxes.

3. METHODOLOGY

Figure 1 shows the overall architecture of the proposed approach, which is based on SSD framework [6]. There are two sub-networks in the proposed framework, including the backbone sub-network and the detection sub-network. In the backbone sub-network, the base convolution layers are generated by the improved architecture based on DenseNets structure [1]. More specific, the backbone sub-network includes stem module, dense blocks, and transition layers. With dense connection as in original DenseNets architecture, the proposed

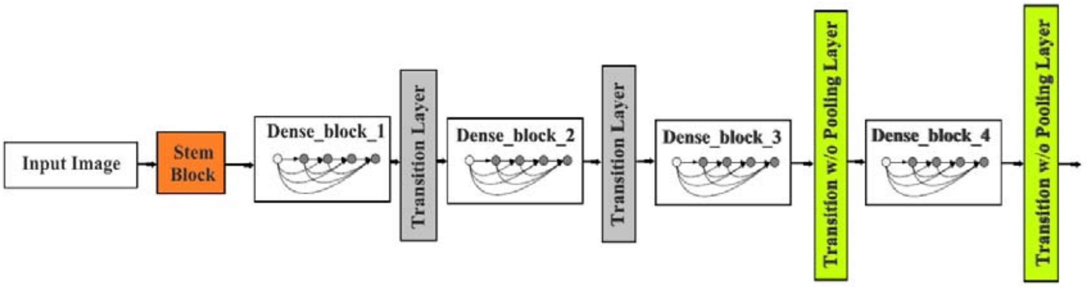

Figure 2: The Overall Architecture of The Backbone Sub-Network.

Table 1: Detail Architecture of The Backbone Sub-Network.

Layer	Name	Kernel Size	Output Size
0	Stem	3×3 conv	75×75×64
1	Dense_block_1	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times 6$	75×75×256
	Transition layer	1×1 conv	75×75×256
		2×2 max pool, stride 2	38×38×256
2	Dense_block_2	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times 8$	38×38×512
	Transition layer	1×1 conv	38×38×512
		2×2 max pool, stride 2	19×19×512
3	Dense_block_3	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times 8$	19×19×768
	Transition w/o pooling layer	1×1 conv	19×19×768
4	Dense_block_4	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times 8$	19×19×1024
	Transition w/o pooling layer	1×1 conv	19×19×1024

approach can achieve higher accuracy with fewer parameters compared with ResNet architecture. In addition, extra convolution layers used for object classification and location regression are added as in original SSD framework. In the detection sub-network, enhanced feature extraction subnet takes convolution layers generated by the backbone sub-network to generate enhanced feature maps by fusing operation, atrous convolution and deconvolution operation. Enhanced feature extraction subnet takes full advantage of the relationship between different feature layers of the backbone sub-network, thus enhancing detection performance of multi-scale pedestrian detection. Furthermore, residual blocks are added before each prediction layer. Residual blocks can reduce the computational cost while the detection accuracy is improved. Details of each sub-network will be explained in the following subsections.

3.1 The Backbone Sub-Network

The backbone sub-network is a variant of the deeply supervised DenseNets [1] architecture. With dense connection in DenseNets architecture, fewer parameters and high accuracy are achieved compared with ResNet architecture. Figure 2 shows the overall architecture of the backbone sub-network. As shown, the backbone sub-network includes a stem block, four dense blocks, two transition layers and two transition w/o pooling layers. Table 1 shows detail architecture of the

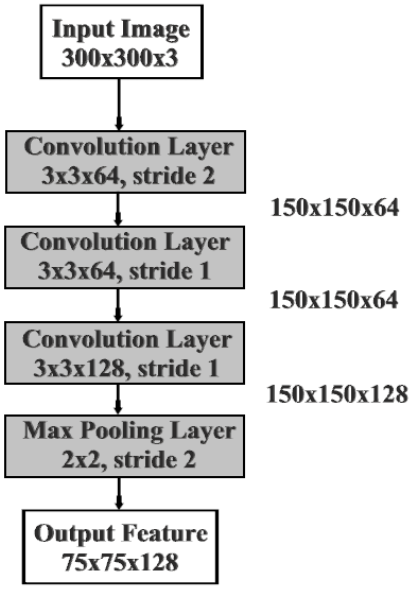

Figure 3: The Architecture of The Proposed Stem Module.

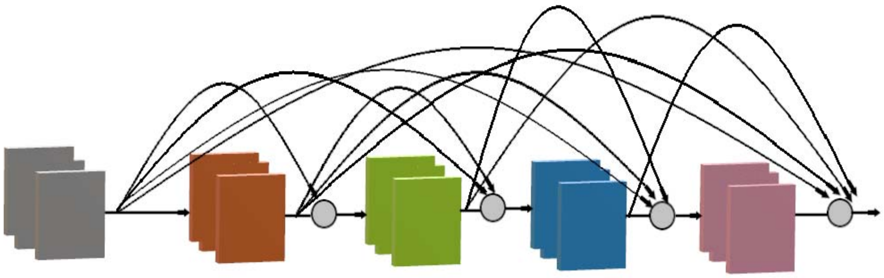

Figure 4: Dense Block in DenseNet Architecture. There are 4 Dense Blocks in The Proposed DenseNet Network.

backbone sub-network. Details of each module will be explained in the following subsections.

3.1.1 Stem Block

Inspired by Inception-v4 architecture [2] and DSOD detector [3], this paper applies an enhanced stem module after the input image. The proposed stem module replaces the convolutional module in the original DenseNets architecture, which uses 7×7 convolution layer with stride = 2 followed by a 3×3 max pooling with stride = 2. Figure 3 shows the architecture of the proposed stem module. As shown, the stem module includes three 3×3 convolution layers followed by a 2×2 max pooling layer. The first convolution layer works with stride = 2 and the other convolution layers are with stride = 1. The proposed stem module can reduce the information loss from raw input images, effectively improve the ability to express features without increasing the computational cost [3].

3.1.2 Dense Block

There are four dense blocks in the backbone sub-network as shown in Table 1. In each dense block, each layer obtains additional inputs from all preceding layers and passes on its own feature-maps to all subsequent layers as shown in Figure 4. Thus, earlier layers in DenseNet can receive additional supervision from the objective function through the skip connections. Although only a single loss function is required on top of the network, all layers including the earlier layers still can share the supervised signals unencumbered. In addition, since each layer receives feature maps from all preceding layers, network can be thinner and more compact.

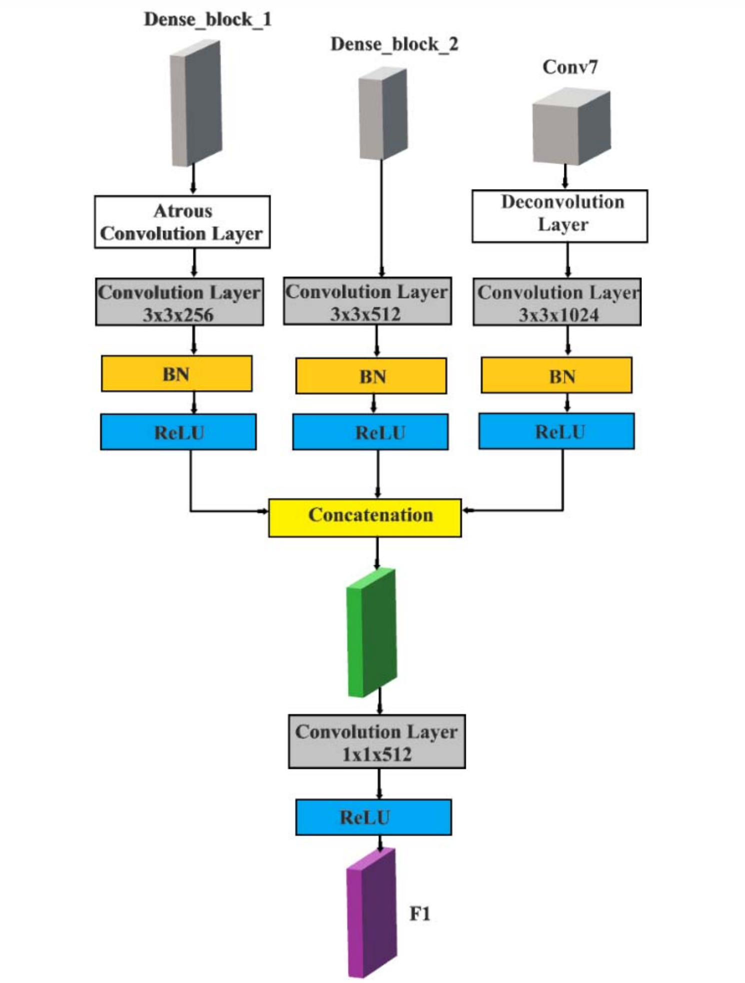

Figure 5: The Structure of The First Enhanced Feature Extraction Subnet.

3.1.3 Transition Layers

In the original DenseNet architecture, each transition layer contains consist of a batch normalization layer and a 1×1 convolutional layer followed by a 2×2 average pooling layer to down-sample the feature maps. The number of dense blocks is fixed at four dense blocks in all DenseNet architectures for maintaining the same scale of outputs. The only way to increase network depth is adding layers inside each block for the original DenseNet. Thus, this paper also uses the transition w/o pooling layer to eliminate the layer limitation of each of the dense blocks of the proposed network architecture, which makes the network deeper and has stronger feature extraction capability, and the final feature map resolution is kept unchanged.

3.1.4 Composite Function

This paper adopts the composite function as in the original DenseNet. The composite function includes three consecutive operations: Batch Normalization (BN) [4], followed by a Rectified Linear Unit (ReLU) [5] and a 3×3 convolution layer.

3.1.5 Growth Rate

The growth rate *k* is the number of 3×3 convolution kernels in the last dense block. Since each dense block is finally connected in a concatenation manner, the feature dimension of the next layer will increase by *k* after each dense block. The larger its value means the greater the amount of information circulating in the network, and the stronger the network performance, but the size and computation of the entire model will also increase. Based on the experiments this paper sets the growth rate *k* at 16 and 32 in the proposed architecture.

3.2 The Detection Sub-Network

The detection sub-network is based on SSD framework [6]. The detection sub-network includes enhanced feature extraction subnet and residual prediction module for object detection on multi-scale feature maps. In the original SSD framework,

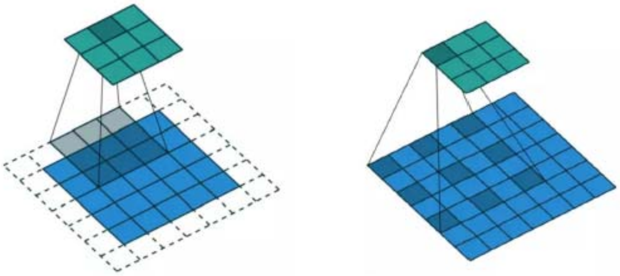

Figure 6: Atrous Convolution Operation with Kernel Size k=3 and Dilation Rate r=1 (Left), and with Kernel Size k=3 and Dilation Rate r=2 (Right).

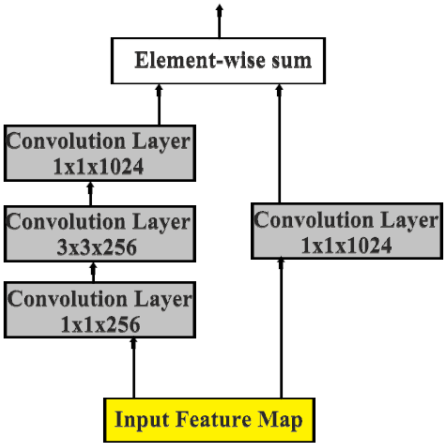
Element-wise sum
Convolution Layer 1x1x1024
Convolution Layer 3x3x256
Convolution Layer 1x1x256
Convolution Layer 1x1x1024
Input Feature Map

Figure 7: The Structure of Each Residual Block.

Conv4_3 layer, Conv7 layer of VGG-16 and the newly added layers, including Conv8_2, Conv9_2, Conv10_2, and Conv11_2 are used for object classification and location regression. In this paper, enhanced multi-scale feature layers are constructed for fast and efficient pedestrian detection. More specific, this paper uses enhanced feature layers, include F1, F2, F3, F4, Conv10_2 and Conv11_2 for object classification and location regression as shown in Figure 1. The feature sizes of the feature layer group composed of these six scales are 38×38, 19×19, 10×10, 5×5, 3×3 and 1×1, which is the same as original SSD. Details of each module will be explained in the following subsections.

3.2.1 Enhanced Feature Extraction Subnet

In original SSD framework, multi-scale feature layers are used for object classification and location regression. The low-level feature maps at shallower convolution layers have higher resolution and smaller receptive field compared with the high-level feature maps at deeper convolution layer. Thus, the low-level feature maps can well represent the detail information such as the texture and edge of the image. This can improve the object location regression, but its weak global semantic features are not conducive to object classification. On the contrary, the high-level feature maps can provide rich semantic information, which can improve the object classification. However, the low resolution of the high-level feature maps is not good for the object location task. This paper proposes four enhanced feature extraction subnets, which takes full advantage of the relationship between different feature layers of the backbone sub-network. In the proposes enhanced feature extraction subnets, the feature maps of the five scales of dense_block_1, dense_block_2, Conv7, Conv8_2, and Conv9_2 are used for generating enhanced feature maps. Figure 5 shows the architecture of the first enhanced feature extraction subnet. As shown in Figure 5, F1 is a feature layer with the size of 38×38×512 generated by the fusion of dense_block_1, dense_block_2 and Conv7. Since dense_block_1 has a higher resolution and contains more details than dense_block_2, atrous convolution is used to down-sample dense_block_1. In addition, deconvolution operation is used to up-sample Conv7 to make the feature map size of the dense_block_1 and Conv7 the same as that of the dense_block_2. A 3×3 convolution layer is then applied on each of the three layers follow by batch normalization layer for normalization processing. The three feature layers are activated before concatenation process. Finally, the convolution operation with convolution kernel size of 1×1 is used to reduce the dimension of the concatenated feature to generate the final enhanced feature layer. The other enhanced feature extraction subnets have similar architecture.

Atrous Convolution

Atrous convolution is a powerful module in dense prediction tasks. It can effectively enlarge the field of view of filters to incorporate larger context without increasing the number of parameters or the amount of computation. Another advantage is that atrous convolution can be conveniently and seamlessly integrated to compute the responses of any layer. In this paper, atrous convolution is used to fuse the low-level feature maps and the high-level feature maps. This can significantly improve the receptive field range of the classification network, which can enhance the model to learn more global information. In the atrous convolution operation, a new super-parameter dilation rate *r* is introduced. Atrous convolution with dilation rate *r* introduces (*r*–*1*) zeros between consecutive filter values, effectively enlarging the kernel size of a $k \times k$ filter to $k' \times k'$ by using equation (1) without increasing the number of parameters or the amount of computation.

$$k' = k + (k - 1)(r - 1) \qquad (1)$$

After performing the atrous convolution, the receptive field *R* is calculated by the following equation:

$$R = \left[2^{\left(\frac{r}{2}+2\right)} - 1\right] \times \left[2^{\left(\frac{r}{2}+2\right)} - 1\right] \qquad (2)$$

Figure 6 (left) illustrates an atrous convolution operation with kernel size *k*=3 and dilation rate *r*=1, which is equivalent to a traditional convolution operation. Figure 6 (right) illustrates an atrous convolution operation with kernel size *k*=3 and dilation rate *r*=2. According to (1) and (2), the new convolution kernel size after the atrous convolution operation is 5×5. Compared with the result in Figure 6 (left), the receptive field *R* is expanded to 7×7 without loss of feature information.

3.2.2 Residual Prediction Subnet

In original SSD framework, a set of convolutional filters is applied at each feature layer to produce a fixed set of predictions. For a feature layer with a size of $w \times h \times n$, a $3 \times 3 \times n$ convolution layer is used to perform the convolution operation to obtain a category score and a shape offset relative to the default box coordinates. This paper adds a residual block before each prediction layer. Figure 7 shows the structure of each residual block. Residual block allows us to apply 1×1 convolution kernel to predict category scores and box offsets. Thus, it can reduce the computational cost while the detection accuracy is improved.

4. EXPERIMENTAL RESULTS

4.1 Dataset and Evaluation Metrics

To demonstrate the effectiveness of the proposed approach, this paper evaluates the proposed method on two of the largest pedestrian detection datasets, including Caltech dataset [7] and CityPersons dataset [8]. The Caltech dataset is one of the most popular datasets for pedestrian detection. It contains 250,000 frames captured from ten hours of urban traffic videos. The training data (set00-set05) consists of six training sets, each with 6–13 one-minute long sequence files, along with all annotation information. The testing data (set06-set10) consists of five sets, again along with all annotation information. The training and testing dataset have different video sequences with respect to the difficulty of pedestrian height, visibility, and aspect ratio. In the proposed experiments, the training images are extracted with one out of every frame. There are 128,419 images for training and 4024 images for testing. CityPersons dataset, which is built on top of the semantic segmentation dataset CityScapes [9], is a more challenging large-scale pedestrian detection dataset with various occlusion levels. This dataset records street views across 18 different cities in Germany with various weather conditions. The dataset includes 5,000 images (2,975 for training, 500 for validation and 1,525 for testing) with 35,000 labeled persons plus 13,000 ignored region annotations. Both bounding box annotations of full bodies and visible parts are provided. For evaluation metrics, this paper follows the standard Caltech evaluation metric [9], that is log-average Miss Rate over False Positive Per Image (FPPI) ranging in [10^{-2}, 10^{0}] (denoted as MR^{-2}, lower is better). MR^{-2} is computed by averaging the Miss Rate at 9 FPPI rates over the range of [10^{-2}, 10^{0}] in log-space.

4.2 Implementation Details

The proposed approach is implemented in Pytorch deep-learning framework with Python interface. The CPU used in all experiments is Intel Core i7-8700, the main memory is 12GB DDR4

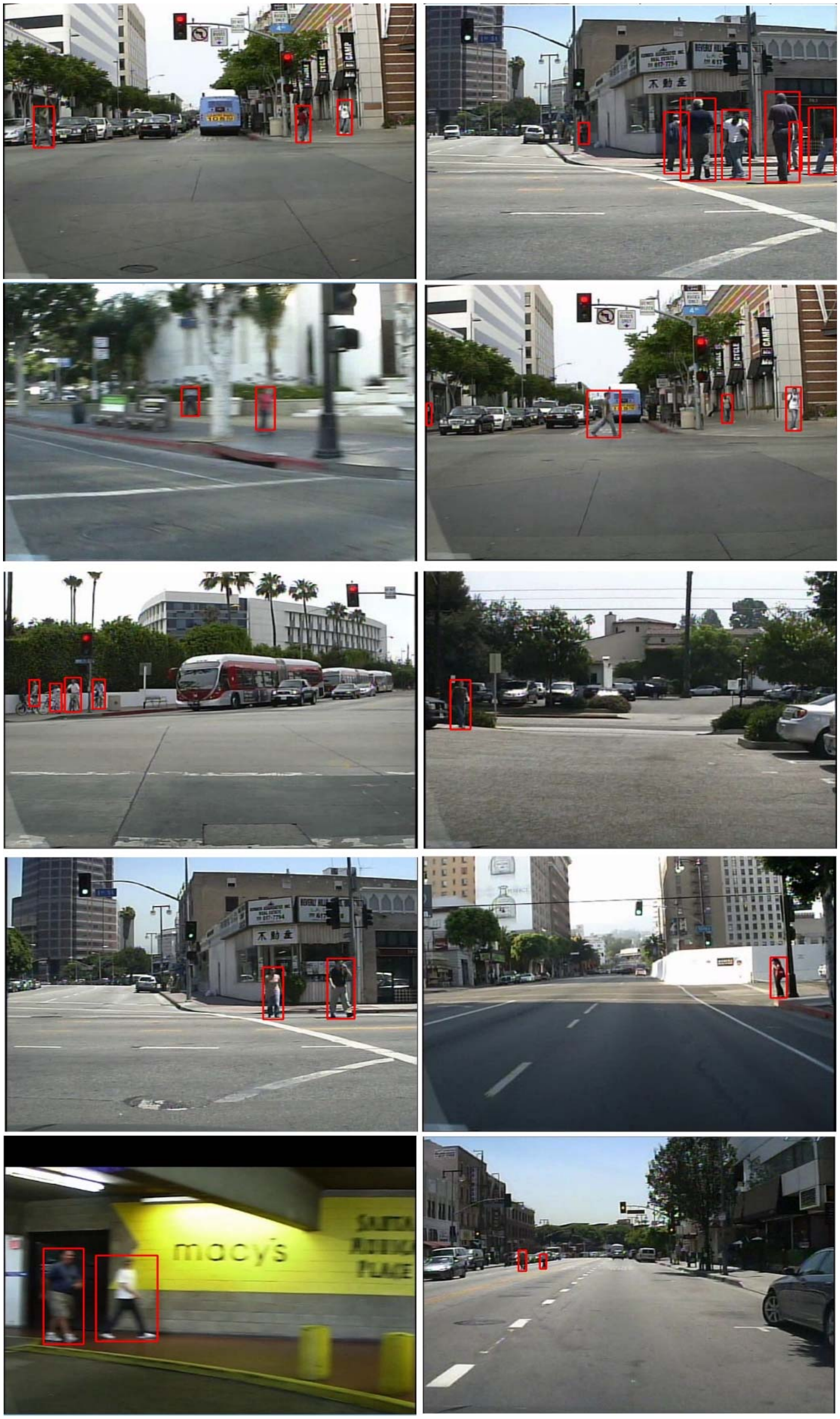

Figure 8: Examples Images of Detection Results on The Caltech Dataset.

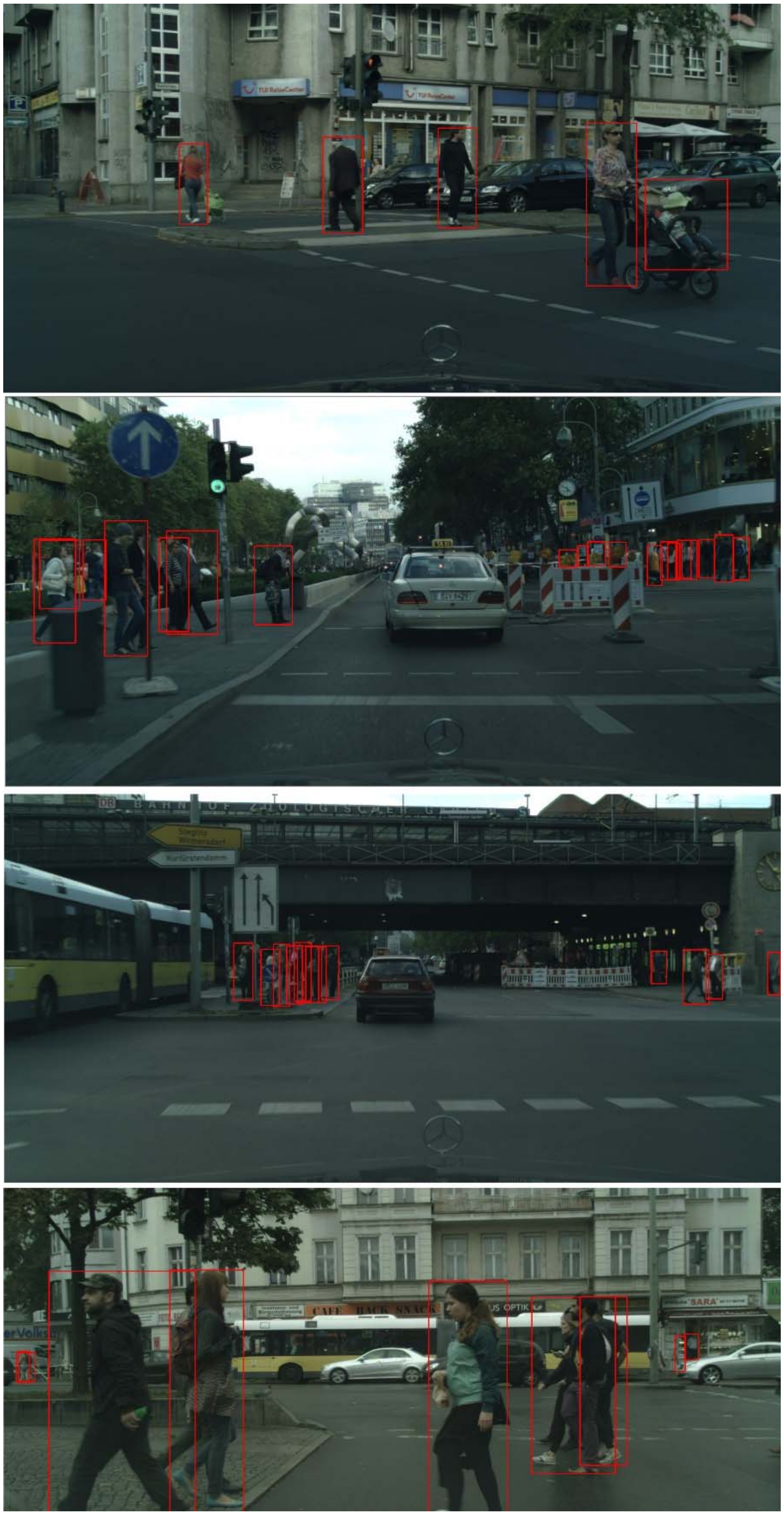

Figure 9: Detection Results on CityPersons Dataset.

RAM, and the GPU is NVIDIA GeForce GTX 1080. In the training phase, all models are trained from scratch. The advantage of training from scratch is that it is not necessary to rely on the pre-training model on classification dataset to initialize the network. Following DSOD, since each scale of enhanced feature extraction subnets is all concatenated from multiple resolutions, L2 normalization technique [10] is adopted to scale the feature norm to 20 on all output. Most of the proposed training strategies follow SSD framework, including scale and aspect ratios for default boxes, loss functions (smooth L1 loss for localization purpose and softmax loss for classification purpose), data augmentation, and so on. This paper also adopts the hard-negative mining technique of SSD so that the ratio between positive and negative samples is at most 3:1, which lead to faster optimization and more stable training.

4.3 Detection Results on The Caltech Dataset

To evaluate the performance of the proposed approach, this paper conducts experiments on the Caltech dataset and compares the detection results of the proposed method with the results of other state-of-the-art methods, including MS-CNN [11], SA-Fast RCNN [12], ADM [13], SDS-RCNN [14], F-DNN+SS [15]. The comparison results are evaluated for pedestrian instances of three cases of the Caltech dataset: Reasonable case (pedestrians with no less than 50 pixels in height), small-scale case (pedestrians with shorter than 50 pixels), and overall case (pedestrians of all scales and occlusions). MS-CNN proposed a multi-scale object detection framework, which detection is performed at multiple output layers. SA-Fast RCNN introduced a new framework with multiple built-in subnetworks which detect pedestrians with scales from disjoint ranges. In addition, outputs from all of the subnetworks were adaptively combined to generate the final detection results that were shown to be robust to large variance in instance scales. ADM proposed an active pedestrian detector that explicitly operates over multiple-layer neuronal representations of the input still image. SDS-RCNN proposed a segmentation infusion network to enable joint supervision on semantic segmentation and pedestrian detection. F-DNN+SS proposed a single shot deep convolutional network for generating all possible pedestrian candidates of different sizes and occlusions, and multiple deep neural networks are used in parallel for further refinement of these pedestrian candidates.

Table 2 shows the detection results of the proposed method and all reference methods. For the reasonable case, the proposed approach achieves a very low log-average miss rate of 7.89%, which is competitive with the best result of SDS-RCNN. However, the proposed approach achieves better result compared with SDS-RCNN in small-scale case. For the small-scale case, the proposed method gets the best result. More specific, the log-average miss rate is improved by 30.53%, 33.23%, 4.74%, 33.32%, and 9.43% compared with MS-CNN, SA-Fast RCNN, ADM, SDS-RCNN, and F-DNN+SS, respectively. For the overall case, the proposed method achieves better result compared with MS-CNN, SA-Fast RCNN, SDS-RCNN, and F-DNN+SS. ADM achieves the best result in the overall case. However, the proposed is faster than ADM. Furthermore, Table 2 shows a comparison report between the proposed method and all reference methods in terms of the computation efficiency. As shown, the inference time of the proposed method surpasses the current state-of-the-art methods for pedestrian detection. More specific, the proposed framework takes 0.11 second to process an image, while the best inference time of the reference methods takes up to 0.21 second. These results show the effectiveness of the proposed method in both detection accuracy and runtime. Figure 8 shows some examples images of detection results. It can observe that most pedestrians, including small-scale pedestrians, can be exactly detected correctly by the proposed approach.

4.4 Detection Results on The CityPersons Dataset

To further evaluate the performance of the proposed method on pedestrian detection, this paper conducts experiments and compares the detection results with the results of other recent methods on the CityPersons, including FRCNN [8], OR-CNN [16], ALFNet [17], and TLL [18]. TLL proposed a novel method integrated with somatic topological line localization and temporal feature aggregation for detecting multi-scale pedestrians. ALFNet introduced a structurally simple but effective module to stack a series of predictors to directly evolve the default anchor boxes of SSD step by step into improving detection results. OR-CNN proposed a new occlusion-aware R-CNN (OR-CNN) to improve the detection accuracy in the crowd. Following the evaluation protocol in CityPersons dataset, all of the proposed models on this dataset are trained on the reasonable training set and evaluated on the reasonable validation set.

Table 3 shows the comparisons with previous state-of-the-art approaches on CityPersons dataset. It can be observed that the proposed method achieves comparable log-average miss rate compared with

Table 2: Detection Results of The Proposed Method and All Reference Methods on The Caltech Dataset.

Method	MR^{-2} (%)			
	Reasonable	Small-scale	Overall	Inference time (s)
MS-CNN [11]	9.95	80.74	60.95	0.4
SA-Fast RCNN [12]	9.68	83.44	62.59	0.59
ADM [13]	8.64	54.95	42.27	0.58
SDS-RCNN [14]	7.36	83.53	61.5	0.21
F-DNN+SS [15]	8.18	59.64	50.29	2.48
Proposed Method	7.89	50.21	48.81	0.11

Table 3: The Comparisons of Detection Results with Previous State-Of-The-Art Methods on CityPersons Dataset.

Method	MR^{-2} (%) (Reasonable validation set)
FRCNN [8]	15.4
OR-CNN [16]	12.8
ALFNet [17]	12.0
TLL [18]	15.5
Proposed Method	13.1

other state-of-the-art approaches. More specific, on the reasonable validation subset, the proposed approach achieves 13.1% of MR^{-2}. Figure 9 shows some detection results on CityPersons dataset. It can be observed that the proposed method performs well in detecting pedestrians with partial occlusion or of multiple scales.

5. CONCLUSIONS

In this paper, a single-stage pedestrian detector is introduced. The proposed approach is based on the SSD framework. The backbone sub-network is based on DenseNets structure with dense connection to achieve higher accuracy and fewer parameters compared with other state-of-the-art structures. The detection sub-network includes enhanced feature extraction subnets to generate enhanced feature maps, which enhancing detection performance of multi-scale pedestrian and residual blocks added before each prediction layer. Extensive experiments demonstrated that the proposed framework is superior in detecting multi-scale pedestrian and achieves comparable or better performance relative to other state-of-the-art methods on pedestrian detection. In future work, this paper will incorporate the proposed design with other single-stage detectors like YOLO and FPN.

REFERENCES:

[1] Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. "Densely connected convolutional networks." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700-4708. 2017.

[2] Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. "Inception-v4, inception-resnet and the impact of residual connections on learning." In *Thirty-first AAAI conference on artificial intelligence*. 2017.

[3] Shen, Zhiqiang, Zhuang Liu, Jianguo Li, Yu-Gang Jiang, Yurong Chen, and Xiangyang Xue. "Dsod: Learning deeply supervised object detectors from scratch." In *Proceedings of the IEEE international conference on computer vision*, pp. 1919-1927. 2017.

[4] Ioffe, Sergey, and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." *arXiv preprint arXiv:1502.03167* (2015).

[5] Glorot, Xavier, Antoine Bordes, and Yoshua Bengio. "Deep sparse rectifier neural networks." In *Proceedings of the fourteenth international conference on*

artificial intelligence and statistics, pp. 315-323. 2011.

[6] Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. "Ssd: Single shot multibox detector." In *European conference on computer vision*, pp. 21-37. Springer, Cham, 2016.

[7] Dollar, Piotr, Christian Wojek, Bernt Schiele, and Pietro Perona. "Pedestrian detection: An evaluation of the state of the art." *IEEE transactions on pattern analysis and machine intelligence* 34, no. 4 (2011): 743-761.

[8] Zhang, Shanshan, Rodrigo Benenson, and Bernt Schiele. "Citypersons: A diverse dataset for pedestrian detection." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3213-3221. 2017.

[9] Cordts, Marius, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. "The cityscapes dataset for semantic urban scene understanding." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213-3223. 2016.

[10] Liu, Wei, Andrew Rabinovich, and Alexander C. Berg. "Parsenet: Looking wider to see better." *arXiv preprint arXiv:1506.04579* (2015).

[11] Cai, Zhaowei, Quanfu Fan, Rogerio S. Feris, and Nuno Vasconcelos. "A unified multi-scale deep convolutional neural network for fast object detection." In *European conference on computer vision*, pp. 354-370. Springer, Cham, 2016.

[12] Li, Jianan, Xiaodan Liang, ShengMei Shen, Tingfa Xu, Jiashi Feng, and Shuicheng Yan. "Scale-aware fast R-CNN for pedestrian detection." *IEEE transactions on Multimedia* 20, no. 4 (2017): 985-996.

[13] Zhang, Xiaowei, Li Cheng, Bo Li, and Hai-Miao Hu. "Too far to see? Not really!—pedestrian detection with scale-aware localization policy." *IEEE transactions on image processing* 27, no. 8 (2018): 3703-3715.

[14] Brazil, Garrick, Xi Yin, and Xiaoming Liu. "Illuminating pedestrians via simultaneous detection & segmentation." In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4950-4959. 2017.

[15] Du, Xianzhi, Mostafa El-Khamy, Jungwon Lee, and Larry Davis. "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.

[16] Zhang, Shifeng, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z. Li. "Occlusion-aware R-CNN: detecting pedestrians in a crowd." In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 637-653. 2018.

[17] Liu, Wei, Shengcai Liao, Weidong Hu, Xuezhi Liang, and Xiao Chen. "Learning efficient single-stage pedestrian detectors by asymptotic localization fitting." In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 618-634. 2018.

[18] Song, Tao, Leiyu Sun, Di Xie, Haiming Sun, and Shiliang Pu. "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.

[19] Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jitendra 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. 2014.

[20] Girshick, Ross. "Fast r-cnn." In *Proceedings of the IEEE international conference on computer vision*, pp. 1440-1448. 2015.

[21] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." In *Advances in neural information processing systems*, pp. 91-99. 2015.

[22] Dai, Jifeng, Yi Li, Kaiming He, and Jian Sun. "R-fcn: Object detection via region-based fully convolutional networks." In *Advances in neural information processing systems*, pp. 379-387. 2016.

[23] Uijlings, Jasper RR, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. "Selective search for object recognition." *International journal of computer vision* 104, no. 2 (2013): 154-171.

[24] Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali 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. 2016.

[25] Nam, Woonhyun, Piotr Dollár, and Joon Hee Han. "Local decorrelation for improved pedestrian detection." In *Advances in neural information processing systems*, pp. 424-432. 2014.

[26] Li, Qiming, Hanzi Wang, Yan Yan, Bo Li, and Chang Wen Chen. "Local co-occurrence selection via partial least squares for pedestrian detection." *IEEE Transactions on Intelligent Transportation Systems* 18, no. 6 (2016): 1549-1558.

[27] Tian, Yonglong, Ping Luo, Xiaogang Wang, and Xiaoou Tang. "Deep learning strong parts for pedestrian detection." In *Proceedings of the IEEE international conference on computer vision*, pp. 1904-1912. 2015.

[28] Zhou, Chunluan, and Junsong Yuan. "Multi-label learning of part detectors for heavily occluded pedestrian detection." In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3486-3495. 2017.

[29] Zhang, Shanshan, Jian Yang, and Bernt Schiele. "Occluded pedestrian detection through guided attention in CNNs." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6995-7003. 2018.

[30] Nguyen, Hoanh. "An efficient deep learning framework for pedestrian detection." In *Journal of Theoretical and Applied Information Technology* 97, no. 21 (2019).

[31] Wang, Xinlong, Tete Xiao, Yuning Jiang, Shuai Shao, Jian Sun, and Chunhua Shen. "Repulsion loss: Detecting pedestrians in a crowd." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7774-7783. 2018.

[32] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." In *Advances in neural information processing systems*, pp. 1097-1105. 2012.