PEDESTRIAN DETECTION BASED ON FAST R-CNN AND BATCH NORMALIZATION

Zhong-Qiu Zhao[a], **Haiman Bian**[a], **Donghui Hu**[a], **Hervé Glotin**[b]
[a] College of Computer Science and Information Engineering
Hefei University of Technology
Hefei, China, 230009
[b] Systems and Information Sciences lab
LSIS CNRS & Univ. of Sud-Toulon Var - La Garde, France
{z.zhao,hudh}@hfut.edu.cn
bhm2164@163.com
h.glotin@gmail.com

ABSTRACT

Most of the pedestrian detection methods are based on hand-crafted features which produce low accuracy on complex scenes. With the development of deep learning method, pedestrian detection has achieved great success. In this paper, we take advantage of a convolutional neural network which is based on Fast R-CNN framework to extract robust pedestrian features for efficient and effective pedestrian detection in complicated environments. We use the EdgeBoxes algorithm to generate effective region proposals from an image, as the quality of extracted region proposals can greatly affect the detection performance. In order to reduce the training time and to improve the generalization performance, we add a batch normalization layer between the convolutional layer and the activation function layer. Experiments show that the proposed method achieves satisfactory performance on the INRIA and ETH datasets.

1 INTRODUCTION

In recent years, pedestrian detection has become an important branch of object detection and has attracted wide attention in computer vision Dalal & Triggs (2005); Dollár et al. (2014; 2009); Viola et al. (2005); Wang et al. (2009). In real life, it is a key problem in automotive safety, video surveillance, smart vehicles and intelligent robotics. Because of the diversity of pedestrian body pose, object occlusions, clothing, lighting change and complicated backgrounds in the video sequence or image, the pedestrian detection is still a challenging task in computer vision.

In pedestrian detection, feature extraction is an important factor to influent the performance. Many features have been proposed by researchers for pedestrian detection, such as Haar-like features Viola & Jones (2004), Integral Channel Features (ICF) Dollár et al. (2009), Histogram of Oriented Gradients (HOG) Dalal & Triggs (2005), Local Binary Pattern (LBP) Mu et al. (2008), Dense SIFT Vedaldi et al. (2009) etc. And the proposed Deformable Part Based Model (DPM) Felzenszwalb et al. (2010) which is based on HOG, has maken a breakthrough in pedestrian detection. However, these features are hand-craft and are considered to be low-level. They use the low-level information while the high-level information is usually very important for pedestrian detection Sermanet et al. (2013).

Recently, with the development of deep learning techniques, deep neural networks have been successfully applied in object recognition tasks Girshick et al. (2014); He et al. (2015); Girshick (2015); Zhao et al. (2014). Deep neural networks can achieve more excellent results due to their capability to learn discriminative features from raw pixels. Many researchers have applied deep learning techniques to pedestrian detection. Sermanet *et al*. Sermanet et al. (2013) proposed a two layers convolutional model which adopts convolutional sparse coding to pre-train convolutional neural network for pedestrian detection. Chen *et al*. Chen et al. (2014) proposed a pre-trained Deep Convolutional Neural Network(DCNN) to learn features from ACF Dollár et al. (2014) detector. These

features are then fed to a SVM classifier. Ouyang *et al*. Ouyang & Wang (2013) proposed a joint deep model that jointly learns four key components in pedestrian detection: feature extraction, deformation handling, occlusion handling and classifier. Li *et al*. Li et al. (2015) proposed a SAF R-CNN which takes scales of pedestrians into account in pedestrian detection. Fast R-CNN Girshick (2015) is one of the best detection approaches based on deep leaning. It determines regions of interest(RoI) in an image using superpixel method, and then extracts features using convolutional neural network model. The extracted features are then passed to softmax layer and bounding box regressor layer which are trained for each object, and finally the pedestrian and its position are detected.

It is well known that training a deep neural network is complicated due to the fact that *internal covariate shift* Ioffe & Szegedy (2015) can degrade the efficiency of training. Internal covariate shift is a change of distribution of the inputs. Which happens during training the feed-forward neural networks, by changing the parameters of a layer. Batch normalization (BN) Ioffe & Szegedy (2015) is a technique to control the distributions of feed-forward neural network activations, thereby reducing internal covariate shift. So deep neural networks trained with batch normalization can converge faster and generalize better. In this paper, we use Fast R-CNN architecture based on batch normalization(BN) to extract robust pedestrian features.

In pedestrian detection task, another important stage is locating the potential windows which may contain pedestrians. As an exhaustive and traditional method, sliding window method has two main shortcomings. First, it needs searching every possible position in an image. Second, it may produce many redundancy windows which affect the quality of detection. To improve the detection efficiency, the approaches of region proposals have been proposed to produce high quality regions, in which the most popular method is Selective Search Uijlings et al. (2013). It can get high quality of regions by using image segmentation. However, its speed is very slow and the detection performance is sensitive to the image quality. And the Selective Search method of extracting candidate windows is infeasible Chen et al. (2014) but cannot provide precise localization of pedestrians. To produce more precise localized candidate windows, we employ EdgeBoxes Zitnick & Dollár (2014) method, which is a fast and effective one Hosang et al. (2014). It is believed that the edge information can precisely describe the object, so the EdgeBoxes method can extract higher quality of the candidate windows than the Selective Search method. Moreover, the EdgeBoxes method actually performs better in terms of the average intersection over union(IoU) across all images in the set Hosang et al. (2015).

In this paper, our main contributions are summarized as follows:

(1) In order to make the pedestrian detection more efficient, we use the EdgeBoxes method which can obtain low-redundancy and high quality of candidate windows.

(2) We use the Fast R-CNN architecture to extract robust features for pedestrian detection. In order to reduce the training time and prevent the gradients that explode or vanish when training the Fast R-CNN, we add the batch normalization(BN) layer into the Fast R-CNN architecture.

The rest of this paper can be organized as follows. In Section 2, we introduce the related work of pedestrian detection and briefly introduce the batch normalization layer. In Section 3, we introduce our pedestrian detection approach. In Section 4, we present our experiment results on two benchmark datasets, and some analyses as well. In Section 5, we conclude our work and discuss the possible advances to our model in the future.

2 Related Work

In this section, we review the related work of two important stages of pedestrian detection, viz. extracting region proposals and extracting features, respectively. And, we also briefly introduce the batch normalization layer.

Region Proposals: In an image, the position of a pedestrian can be anywhere and its size can be arbitrary, so it is necessary to search the whole image to localize the pedestrians. Traditional methods employ sliding window method to find the possible locations on the whole image. The sliding window method can get almost all potential locations through exhaustive search, but it is computational and may produce many redundancy windows. Some region proposals methods were proposed to overcome this problem Hosang et al. (2015), in which the most popular method is

Selective Search Uijlings et al. (2013). It is a combination of exhaustive search and segmentation method. Wang *et al*. Wang et al. (2015) proposed a region proposal fusion algorithm to fusion the Selective Search and BING Cheng et al. (2014) method. Nam *et al*. Nam et al. (2014) proposed LDCF method to extract regions for pedestrian detection. The EdgeBoxes method has been shown as a state-of-the-art region proposal system Zitnick & Dollár (2014); Hosang et al. (2014).

Extracting features: Many models based on hand-crafted features have been utilized for pedestrian detection Dalal & Triggs (2005); Dollár et al. (2009); Wang et al. (2009); Viola & Jones (2004); Mu et al. (2008). Wang *et al*. Wang et al. (2009) utilized the combination of HOG and LBP to handle the partial occlusion of pedestrian. Felzenswalb *et al*. Mu et al. (2008) further improved the detection performance by combining the HOG with a deformable part model. Dollár *et al*. proposed the Integral Channel Features(ICF) Dollár et al. (2009) and Aggregated Channel Features(ACF) Dollár et al. (2014) which efficiently extract histograms and haar features. Chen *et al*. Chen et al. (2013) used a multi-order context representation to take advantage of co-occurrence contexts of different objects. Deep learning models can be trained end-to-end to extract robust features from the given images. Sermanet *et al*. Sermanet et al. (2013) proposed a convolutional neural network model which has two convolutional layers pre-trained by sparse coding. Ouyang *et al*. Ouyang & Wang (2013) proposed a joint deep model which contains a deformation layer to model mixture poses information. Tian *et al*. Tian et al. (2015) jointly utilized semantic tasks to optimize pedestrian detection. More recently, some excellent works have been successfully applied in object detection Girshick et al. (2014); He et al. (2015); Girshick (2015), and the performance of pedestrian detection is largely improved Chen et al. (2014); Li et al. (2015); Wang et al. (2015). Fast R-CNN Girshick (2015) is one of the best detection approaches among them. It consists of two steps for object detection, in which the first step is to use non-deep learning methods to extract thousands of region proposals, and the second step is to use convolutional neural networks classifiers to classify categories at those locations.

Batch Normalization: Training a deep neural network is complicated due to the fact that changing the parameters of a layer will affect the distribution of the inputs to the succedent layers. As a result, the succedent layers are continually adapted to fit the shifting input distribution and unable to learn effectively. This phenomenon is called *Internal Covariate Shift* (ICS) Ioffe & Szegedy (2015). The ICS slows down the course of training and needs careful parameter initialization. Batch normalization(BN) Ioffe & Szegedy (2015) is a technique to accelerate training and to improve generalization capability. Its basic idea is to standardize the internal representations inside the network to make the network converging faster and generalizing better, inspired by the way to whiten the network input to improve performance. Batch normalization layer can be applied to convolutional neural networks directly between the convolutional layer and the activation function layer, for example ReLU etc. It uses the local mean and variance computed over the minibatch x, and then corrects with a learned variance and bias term, namely γ and β:

$$BN(x;\gamma,\beta) = \beta + \gamma\frac{x - E[x]}{(Var[x] + \epsilon)^{1/2}} \tag{1}$$

where ϵ is a small positive constant to improve numerical stability.

3 Our Approach

In this section, we will introduce the details of our approach for pedestrian detection. Firstly, we will introduce the proposal generation algorithm. Then, we will describe the pedestrian detection architecture. Finally, we will briefly introduce the whole pedestrian detection process.

3.1 Pedestrian Proposals

We utilize the EdgeBoxes as our proposal generation algorithm. The basic idea of EdgeBoxes is that it generates and scores the proposal based on the edge map of the given image.

Specifically, firstly, a structured edge detector is utilized to generate an edge map where each pixel contains a magnitude and orientation information of the edge. Secondly, the edges are grouped together by a greedy algorithm. Then, the affinity between two edge groups is computed. This is the input to the next step for the score computation. Thirdly, for a given bounding box, the score of the

bounding box is computed based on the edge groups entirely inside the box. Finally, as the Edge-Boxes detector may generate many overlapping candidate pedestrian windows, the non-maximal suppression is used to filter out these overlapping bounding boxes by the following criterion:

$$OverlapArea = \frac{(IntersectionArea)}{(UnionArea)} \tag{2}$$

If two bounding boxes overlap by more than 50%, then the bounding box with the highest score is selected.

Compared with the Selective Search which is the most popular proposal method, EdgeBoxes is much faster. For EdgeBoxes, the average runtime is about 0.3 seconds per image while Selective Search is about 10 seconds per image. In the case of accuracy on the PASCAL VOC 2007 dataset, the mAP of Selective Search is 31.7% which is slightly smaller than the mAP of EdgeBoxes 31.8% Hosang et al. (2014), and on the INRIA dataset, the recall rates of Selective Search and EdgeBoxes are 23% and 93% respectively by our experiment. In brief, the EdgeBoxes can not only reduce the runtime complexity but also increase the accuracy. Therefore, we choose the EdgeBoxes as our proposal generation algorithm.

3.2 The Architecture for Pedestrian Detection

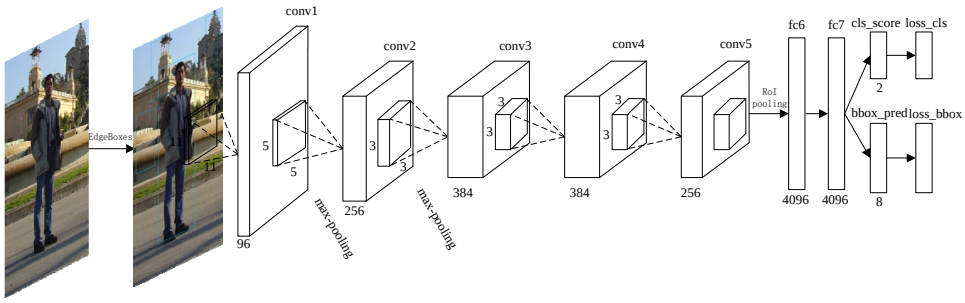

Figure 1: The architecture of our proposed pedestrian detection model, where the batch normalization layer is placed between any one convolutional layer and the succedent activation function layer (such as ReLU layer, and details can be seen in Figure 2).

The architecture of our proposed Fast R-CNN based on batch normalization method is illustrated in Figure 1. It has 7 layers. As batch normalization layer can normalize each scalar feature independently layer by layer, it avoids the gradient vanish and gradient explode problem. Batch normalization can lead to a faster convergence during the training and can improve the generalization performance as a regularization technique. In our proposed architecture, the batch normalization layer is placed between each convolutional layer and the succedent ReLU layer, as illustrated in Figure 2.

In the first convolutional layer, there are 96 kernels of size 11×11 with a stride of 4 pixels and in the following max-pooling layer, the kernel size is 3×3. The second convolutional layer takes as input the output of the first convolutional layer, and there are 256 kernels of size 5×5 with a stride of 2 pixels. In the following max-pooling layer, the kernel size is the same as the first layer. For the next two (the third and the fourth) convolutional layers, the corresponding pooling layers are omitted and both contain 384 kernels. In the fifth convolutional layer, there are 256 kernels and the following

Figure 2: The first five convolutional layers of our proposed model. The batch normalization (BN) layer is placed between each convolutional layer (Conv) and the succedent ReLU layer. In the first two convolutional layers, the max-pooling layer is placed to follow the ReLU layer.

pooling layer is a region of interest (RoI) pooling layer. The RoI pooling layer is utilized to pool the feature maps of each input object proposal into a fixed-length feature vector which is then fed into the fully connected layers. The next two layers are fully connected layers, both of which contain 4096 nodes. At the end of the architecture there are two output layers which produce two output vectors for per object proposal. Specifically, one is a softmax layer, which outputs classification scores over K object classes plus a "background" class. The other is a bounding-box regressor layer, which outputs four real-valued numbers for each of the K object classes. These $4\times$K values encode refined bounding boxes for each class.

3.3 Pedestrian Detection Process

The entire pedestrian detection process can be summarized as follows:

(1) Run the EdgeBoxes algorithm to generate region proposals.

(2) Input the whole image and a number of region proposals to our proposed model.

(3) Extract the features of region proposals with convolutional neural networks.

(4) Locate and verify pedestrian in each region proposal with extracted features by softmax classifier and bounding-box regressor.

The overall framework is also presented in Figure 3.

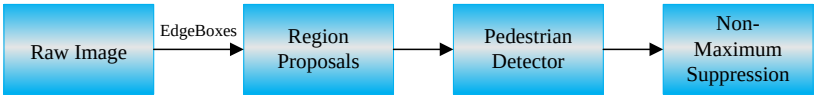

Figure 3: The overview of our pedestrian detection system. (1) Use EdgeBoxes to generate region proposals. (2) Pass the region proposals to the pedestrian detector which is our proposed model. The detector will produce softmax scores for each proposal bounding box. (3) For each pedestrian class, use the non-maximum suppression (NMS) independently to greedily merge the overlapped proposals.

4 Experiments and Analyses

In this section, we evaluate our proposed pedestrian detection method on two popular benchmark datasets, INRIA and ETH. We follow the evaluation protocols proposed in Dollár et al. (2012). The log average miss rate is used to summarize the detector performance. It is computed by averaging the miss rate at 9 FPPI (false positives per image) rates (0.05, 0.10, 0.20, 0.30, 0.40, 0.50, 0.64, 0.80, 1) which are evenly spaced in the log-space ranging from 10^{-2} to 10^{0}. In one test image, if a detected bounding box BB_{dt} can significantly cover most area of the body window BB_{gt} of the ground-truth pedestrian, the detection window is considered to properly detect the pedestrian.

$$\alpha_0 = \frac{area(BB_{dt} \cap BB_{gt})}{area(BB_{dt} \cup BB_{gt})} \geqslant \theta \tag{3}$$

where α_0 is the area of overlap between BB_{dt} and BB_{gt}, and θ is the threshold which is set to be 0.5. And the experiments are run on an Ubuntu 14.04 64 bit system with a single 4GB memory NVIDIA GeForce GTX 980 GPU.

4.1 INRIA Dataset

The INRIA dataset is a very popular dataset for pedestrian detection. The annotations of pedestrians are of high quality in diverse conditions and poses. The details of INRIA dataset are listed in Table 1.

Our proposed model is implemented based on the publicly Caffe platform Jia et al. (2014) and the CaffeNet model is taken as our initial model which has been pre-trained on the ImageNet dataset. In

Table 1: The details of INRIA dataset

Statistic	Training	Testing
Positive images	614	288
Negative images	1218	453
Annotated windows	1237	589

order to use the Fast R-CNN method, we replace the last max-pooling layer of the CaffeNet with the RoI pooling layer to pool the feature maps of each region proposal into a fixed resolution, *i.e.* 6×6. We also replace the final fully connected layer and softmax layer with two sibling fully connected layers, i.e. softmax layer and bounding-box regressor layer.

In order to expand positive example set, we map each region proposal to the ground-truth which has a high intersection over union (IoU). If the IoU is lager than 0.5, then we will label the selected region proposal as a positive sample for pedestrian class. The rest region proposals, whose IoUs are smaller than 0.1, are regarded as negative instances. The proposed model is trained with the Stochastic Gradient Descent (SGD) method with the momentum of 0.9 and the weight decay factor of 0.0005. Since batch normalization layer is added, a large learning rate is used in our experiment. We set the initial global learning rate as 0.01 and the dropout ratio as 0.1 in the fc6 and fc7 layers. The whole process of training is 40000 iterations, the training time of our model (FRCNN+BN) is 0.094 seconds per iteration and that of Fast R-CNN is 0.153 seconds per iteration. The average time to process an image using our model is about 0.5 seconds, while the average time of the Fast R-CNN is about 0.6 seconds.

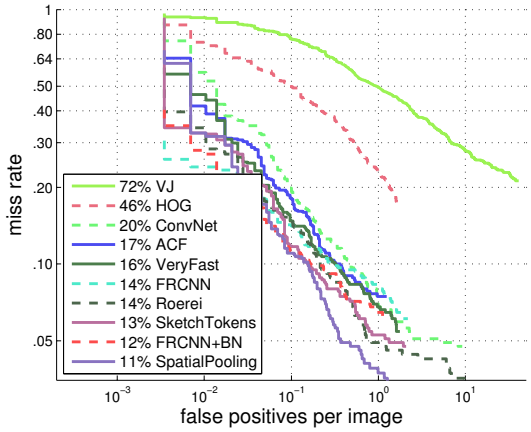

Figure 4: Comparison of miss rate versus false positives per image between different methods on the INRIA dataset. Our model is denoted as FRCNN+BN. We obtain the miss rate of 12% at 0.1 FPPI.

The overall experimental results are reported with the DET curves (miss rate versus FPPI) in Figure 4. We compare our proposed model with existing methods, including two classical methods VJ Viola & Jones (2004), HOG Dalal & Triggs (2005) and other state-of-the-art methods such as ConvNet Sermanet et al. (2013), ACF Dollár et al. (2014), VeryFast Rodrigo et al. (2012), FRCNN Girshick (2015), Roerei Rodrigo et al. (2013), Sketch Tokens Joseph et al. (2013) and SpatialPooling Paisitkriangkrai et al. (2014). We also give the digitized results of the miss rate at 0.1 FPPI.

From Figure 4, we can see that our method outperforms most of the state-of-the-art methods. The miss rate of our method is 12%, which has an improvement of 8% over ConvNet and an improvement of 2% over FRCNN, but a slight decrease over the SpatialPooling method.

4.2 ETH DATASET

The ETH dataset derives from the binocular vision of pedestrian dataset. It is obtained by a pair of on-board camera and it gives the annotations of pedestrians. The set01, set02 and set03 are currently the most used three subsets. The details of these three subsets can be seen in Table 2.

Table 2: The details of ETH dataset

Statistic	set01	set02	set03
Positive images	999	446	354
Negative images	0	5	0
Annotated windows	8466	3472	2225

In order to evaluate the generalization performance of our proposed method, we train our proposed method on the INRIA dataset, and then apply the trained model to the ETH dataset. The performance evaluation is the same as that on the INRIA dataset. The average time to process an image using our model (FRCNN+BN) is about 0.4 seconds, while the average time of the Fast R-CNN is about 0.5 seconds. We compare our model with other methods, such as ConvNet Sermanet et al. (2013), JointDeep Ouyang & Wang (2013), FRCNN Girshick (2015), SDN Luo et al. (2014), TA-CNN Tian et al. (2015), VJ Viola & Jones (2004), HOG Dalal & Triggs (2005), ACF Dollár et al. (2014), VeryFast Rodrigo et al. (2012). The comparison results are shown in Figure 5, from which we can see that our method outperforms VJ, HOG, VeryFast, ACF, ConvNet, FRCNN and JointDeep but underperforms SDN, SpatialPooling and TA-CNN methods.

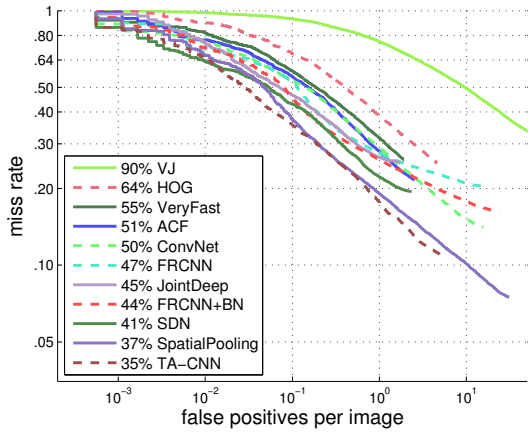

Figure 5: Comparison of miss rate versus false positives per image between different methods on the ETH dataset. Our method is denoted as FRCNN+BN. We obtain the miss rate of 44% at 0.1 FPPI.

From Figure 5, we can also see that the performance on the ETH dataset is worse than that on the INRIA dataset. One reason may lie in that the image quality of ETH dataset is much worse than that of the INRIA dataset, and the image resolution of the ETH dataset is lower. Another reason may be that we use an external dataset to train the model for the ETH dataset.

Finally, we also give some of our detection examples in Figure 6.

5 CONCLUSIONS

In this paper, we propose a novel deep pedestrian detection method based on the Fast R-CNN framework. To reduce the training time and improve the generalization performance, we add the batch

normalization layer between the convolutional layer and activation function layer. To further improve the pedestrian detection speed and accuracy, we use the EdgeBoxes algorithm to remove the redundant windows with poor quality. The experiments show that the proposed method can achieve satisfactory performance comparable with other state-of-the-art methods on two popular pedestrian detection benchmark datasets, INRIA and ETH.

As the image quality can affect the final detection performance, using our proposed method to achieve satisfactory detection performance on low resolution image is still an open issue.

(a) (d) (b) (e) (c) (f)

Figure 6: Detection examples from INRIA dataset (a, b, c) and ETH dataset (d, e, f).

ACKNOWLEDGMENTS

This research was supported by the National Natural Science Foundation of China (No.61672203 & 61375047), the Program for Changjiang Scholars and Innovative Research Team in University of the Ministry of Education of China (No.IRT13059), and the Fundamental Research Funds for the Central Universities of China.

REFERENCES

Guang Chen, Yuanyuan Ding, Jing Xiao, and Tony Han. Detection evolution with multi-order contextual co-occurrence. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1798–1805, 2013.

Xiaogang Chen, Pengxu Wei, Wei Ke, Qixiang Ye, and Jianbin Jiao. Pedestrian detection with deep convolutional neural network. In *Computer Vision-ACCV 2014 Workshops*, pp. 354–365, 2014.

Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin, and Philip Torr. Bing: Binarized normed gradients for objectness estimation at 300fps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3286–3293, 2014.

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pp. 886–893, 2005.

Piotr Dollár, Zhuowen Tu, Pietro Perona, and Serge Belongie. Integral channel features. *British Machine Vision Conference*, 2:5, 2009.

Piotr Dollár, 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(4): 743–761, 2012.

Piotr Dollár, Ron Appel, Serge Belongie, and Pietro Perona. Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1532–1545, 2014.

Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, 2015.

Ross Girshick, 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.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, 2015.

Jan Hosang, Rodrigo Benenson, and Bernt Schiele. How good are detection proposals, really? *arXiv preprint arXiv:1406.6962*, 2014.

Jan Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele. What makes for effective detection proposals? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(4):814–830, 2015.

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

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM International Conference on Multimedia*, pp. 675–678, 2014.

J. Lim Joseph, Zitnick C. Lawrence, and Piotr Dollár. Sketch tokens: A learned mid-level representation for contour and object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3158–3165, 2013.

Jianan Li, Xiaodan Liang, ShengMei Shen, Tingfa Xu, and Shuicheng Yan. Scale-aware fast r-cnn for pedestrian detection. *arXiv preprint arXiv:1510.08160*, 2015.

Ping Luo, Yonglong Tian, Xiaogang Wang, and Xiaoou Tang. Switchable deep network for pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 899–906, 2014.

Yadong Mu, Shuicheng Yan, Yi Liu, Thomas Huang, and Bingfeng Zhou. Discriminative local binary patterns for human detection in personal album. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

Woonhyun Nam, Piotr Dollár, and Joon Hee Han. Local decorrelation for improved pedestrian detection. In *Advances in Neural Information Processing Systems*, pp. 424–432, 2014.

Wanli Ouyang and Xiaogang Wang. Joint deep learning for pedestrian detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2056–2063, 2013.

Sakrapee Paisitkriangkrai, Chunhua Shen, and Anton van den Hengel. Strengthening the effectiveness of pedestrian detection with spatially pooled features. In *Computer Vision–ECCV*, pp. 546–561. 2014.

Benenson Rodrigo, Mathias Markus, Timofte Radu, and Van Gool Luc. Pedestrian detection at 100 frames per second. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2903–2910, 2012.

Benenson Rodrigo, Mathias Markus, Tuytelaars Tinne, and Van Gool Luc. Seeking the strongest rigid detector. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3666–3673, 2013.

Pierre Sermanet, Koray Kavukcuoglu, Soumith Chintala, and Yann LeCun. Pedestrian detection with unsupervised multi-stage feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3626–3633, 2013.

Yonglong Tian, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Pedestrian detection aided by deep learning semantic tasks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5079–5087, 2015.

Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

Andrea Vedaldi, Varun Gulshan, Manik Varma, and Andrew Zisserman. Multiple kernels for object detection. In *IEEE International Conference on Computer Vision*, pp. 606–613, 2009.

Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

Paul Viola, Michael J Jones, and Daniel Snow. Detecting pedestrians using patterns of motion and appearance. *International Journal of Computer Vision*, 63(2):153–161, 2005.

Bin Wang, Sheng Tang, Ruizhen Zhao, Wu Liu, and Yigang Cen. Pedestrian detection based on region proposal fusion. In *Multimedia Signal Processing (MMSP)*, pp. 1–6. IEEE, 2015.

Xiaoyu Wang, Tony X Han, and Shuicheng Yan. An hog-lbp human detector with partial occlusion handling. In *IEEE International Conference on Computer Vision*, pp. 32–39, 2009.

Zhong-Qiu Zhao, Bao-Jian Xie, Yiu-ming Cheung, and Xindong Wu. Plant leaf identification via a growing convolution neural network with progressive sample learning. In *Computer Vision–ACCV 2014*, pp. 348–361. 2014.

C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *Computer Vision–ECCV*, pp. 391–405. 2014.