Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 3518959, 10 pages
https://doi.org/10.1155/2018/3518959

Research Article

An Efficient Pedestrian Detection Method Based on YOLOv2

Zhongmin Liu,[1,2] **Zhicai Chen**,[1,2] **Zhanming Li**,[1,2] **and Wenjin Hu**[3]

[1] *College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China*
[2] *Key Laboratory of Gansu Advanced Control for Industrial Processes, Lanzhou University of Technology, Lanzhou 730050, China*
[3] *College of Mathematics and Computer Science, Northwest Minzu University, Lanzhou 730000, China*

Correspondence should be addressed to Zhicai Chen; 321888594@qq.com

Received 5 August 2018; Revised 7 November 2018; Accepted 21 November 2018; Published 31 December 2018

Academic Editor: Simone Bianco

In recent years, techniques based on the deep detection model have achieved overwhelming improvements in the accuracy of detection, which makes them being the most adapted for the applications, such as pedestrian detection. However, speed and accuracy are a pair of contradictions that always exist and have long puzzled researchers. How to achieve the good trade-off between them is a problem we must consider while designing the detectors. To this end, we employ the general detector YOLOv2, a state-of-the-art method in the general detection tasks, in the pedestrian detection. Then we modify the network parameters and structures, according to the characteristics of the pedestrians, making this method more suitable for detecting pedestrians. Experimental results in INRIA pedestrian detection dataset show that it has a fairly high detection speed with a small precision gap compared with the state-of-the-art pedestrian detection methods. Furthermore, we add weak semantic segmentation networks after shared convolution layers to illuminate pedestrians and employ a scale-aware structure in our model according to the characteristics of the wide size range in Caltech pedestrian detection dataset, which make great progress under the original improvement.

1. Introduction

The category of pedestrian detection is subordinate to the category of the target detection, which is a very popular research subject for its importance in many fields of computer vision. Quite a few applications are inseparable from the pedestrian detection technology, such as the intelligent surveillance system and the autopilot system. Despite the great improvements in accuracy, the task of pedestrian detection is still a great challenge with various difficulties that requires more meticulous design and optimization. Over the past few decades, pedestrian detection methods have adopted a variety of different measures [1–4]. Some of the methods are aimed at increasing the speed of detecting [1, 3]. On the contrary, the other methods have focused on the accuracy [5, 6]. While with the rapid development of the computer hardware and the software, the deep learning began to set off heat waves. Especially, Convolutional Neural Networks (CNN) have appeared as the state-of-the-art technology in the accuracy of a host of computer vision tasks. And methods based on the deep learning usually precede the previous traditional ones by a wide margin in the comprehensive performance.

When the deep network is employed in the task of pedestrian detection, a host of measures have analogous computation pipelines. For the most of the detection frameworks, they usually proceed in two phases. In the first stage, utilizing the original image in the pixel level, they are designed to extract the high-level spatial properties or the high-level features in order to gain some regions of interest. Then, the features of those regions are fed into a classifier or several classifiers that judge if such a region describes a pedestrian. Furthermore, some multiscale measures might be normally adopted to detect the objects at distinct yardsticks for improving detection performance. The pipeline mentioned above regards the task of pedestrian detection as a sort of classification problem. This is also a conventional pipeline. In this paper, we will introduce the YOLOv2 [7] network as our basic framework. Differing from the conventional pipelines, it regards the detection task as a regression problem with the higher speed and accuracy.

1.1. Previous Work. The study of pedestrian detection has gone through several decades, and all sorts of the technologies have been employed in the pedestrian detection, many of which have had a significant impact. Some measures are aimed at improving the basic features utilized [8–10], while others are intended to optimize the algorithms for detection [5, 11]. Meanwhile, some of the techniques will incorporate Deformable Parts Models [12] or take advantage of the context [12, 13].

There are two significantly important tasks in the field of pedestrian detection. One is the contribution of Dollar et al. [14]. They exploit a toolbox and a benchmarking dataset for the public. Accordingly, a number of existing or forthcoming methods could be evaluated without prejudice. And Benenson et al. [8] brought forward a paper that assessed the comprehensive performance of multifarious features and techniques. The other is of Benenson et al. [3] who proposed the fastest technique, reaching a speed of more than 100 fps, which increases the speed of the pedestrian detection.

Since the deep learning entered the field of research, pedestrian detection has been greatly improved in its accuracy [5, 13, 15]. Nevertheless, their running time has been a bit slower, approximately a few seconds every image or even more slowly. In addition, there are several impressive methods employed in the deep network.

The method, ConvNet [16], uses the convents for the pedestrian detection. It will employ the means of convolutional sparse coding to initialize each layer of the network at the beginning and then finely tune the whole network subsequently for the final detection. RPN-BF [17] applies Region Proposal Networks (RPN) proposed in general detector Raster R-CNN [18] to generate the candidate boxes and the high-resolution convolutional feature maps as well as the confidence scores. And then it employs RealBoost algorithm by using the information obtained to shape the Boosted Forest classifier. The perfect fusion of the two stages makes a good performance test for pedestrians. F-DNN [19] is proposed for the fast and robust pedestrian detection with a deep fusion neural network. This architecture is able to concurrently process several networks by improving the processing speed. In order to bring all the possible resigns, a detector is trained to employ the deep convolutional network. For addressing a host of false positives introduced, it introduces a strategy based on the fusion technique to gain the final confidence scores. Furthermore, the technique integrates semantic segmentation network into the trunk network to reinforce the pedestrian detector.

1.2. Contributions. In the application of deep learning, the design of the structure and the setting of the parameters are normally pivotal to get the good results in accuracy. Subtle changes in parameters and structures may result in the quite different results in the overall performance of the system. In the following, we intend to build upon the work of Redmon et al. [7], attentively analyze and revise their models, and then apply them to the pedestrian detection. We employ the clustering algorithm mentioned in his paper to preprocess the training dataset to get the initial candidate boxes. We introduce certain technology for different data, such as multiscale, semantic fusion, and scale-aware. Experiments show that our network used in the pedestrian detection could get better results.

2. Based Detector

YOLOv2, an improved version of YOLO [20], is a detection model with the superior performance applied to the general detection tasks. YOLOv2 could run at the different sizes employing a novel as well as the multiscale training technique. Meanwhile it could offer a rather good trade-off between speed and accuracy, being able to outperform advanced techniques like Faster R-CNN, SSD and so on but still run faster than those all. The YOLOv2 network integrates the extraction of the candidate boxes, the feature extraction, the target classification, and the target location into a single deep network. That enables end-to-end training and transforms the traditional detection problem into a regression problem. For achieving an efficient and accurate pedestrian detection, we introduce the general detector, YOLOv2, as the basic framework of our pedestrian detection model, and then make some modifications in the structure and the parameters of the network, adapting better for the pedestrian. For the convenience of description, we name our model YOLO Based Pedestrian Detection, called Y-PD for short, in this paper.

2.1. Detection Algorithm. The Y-PD model subdivides the image into a $M \times N$ grid, and each grid will detect an object if the center of this object falls into that grid cell. Every grid will be given *the B* initial bounding boxes of different specifications. Then get B predicted bounding boxes (x, y, w, h) and confidence scores $Conf(Object)$ defined as (1) for corresponding boxes through the deep convolutional network:

$$Conf\left(Object\right) = P\left(Object\right) \cdot Iou_{pred}^{truth} \tag{1}$$

The score $Conf(Object)$ is meant to the probability of that class felled into the box and the degree of fitting between the object and predicted bounding boxes. $P(Object)$ denotes that if they contain objects in this grid cell,they can be defined as follows:

$$P\left(Object\right) = \begin{cases} 0, & contain\ objects \\ 1, & not\ contain\ objects \end{cases} \tag{2}$$

Ioutruth pred shown in (3) is the ratio of the union and intersection of the ground truth and the predicted box:

$$Iou_{pred}^{truth} = \frac{area\left(box\left(truth\right) \cap box\left(pred\right)\right)}{area\left(box\left(truth\right) \cup box\left(pred\right)\right)} \tag{3}$$

After getting those predicted boxes, Y-PD will employ a nonmaximum suppression algorithm (*NMS*) whose effect is shown in Figure 1 to eliminate the most of the redundant predicted bounding boxes in order to reduce the difficulty of the network learning. And then it deals with the remaining

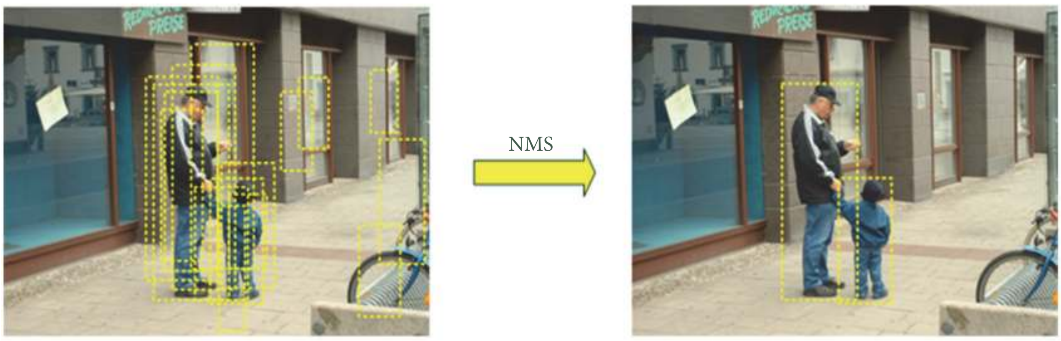


Figure 1: Nonmaximum suppression algorithm.

predicted bounding boxes through the deep convolutional network and obtains the corresponding conditional class probability $Conf(person \mid Object)$, which depends on the mesh cell that contains an object. Then we get the individual bounding box confidence prediction $Conf(person)$ that is defined as follows:

$$\begin{aligned} Conf\left(person\right) &= P\left(person \mid Object\right) \cdot Iou_{pred}^{truth} \\ &= P\left(Person\right) \cdot Iou_{pred}^{truth} \end{aligned} \tag{4}$$

For an input image, the output predictions will be encoded as a $M \times M \times (B * 5 + 1)$ tensor. 5 represents $(x, y, w, h, Conf(person \mid Object))$, while 1 stands for the confidence of a single class, $Conf(person)$.

2.2. Network Architecture. The YOLOv2 network shown in Figure 2 is designed to detect the general object, whose design idea is similar to the Regions Proposal Network (RPN). This network removes the fully connected layer and employs the convolutional network to predict the offset of the bounding box and the confidence. However, its performance in the task of pedestrian detection remains to be raised to a higher level.

Figure 3 is the network framework of our model Y-PD. We can easily find the differences between YOLOv2 and Y-PD. Our model has 23 convolution layers, 6 max pooling layers, 3 reorganization layers, and 1 fusion layer. First, in order to meet the size requirements for the subsequent reorganization, we change the input size from 416×416 to 448×448. Second, aggregating feature maps from multiple levels has been proved to be useful and important in many computer vision tasks [21, 22] for their abilities to collect the rich hierarchical representations. So we add the pass-through layer from one layer to two layers and extract the feature maps from max4 and con5_5, respectively. This technique is capable of making full use of the lower level information and the higher level information, which could increase the accuracy of detection and location. After that, we reorganize the two pass-through layers, making both of them the same size as conv6_7, so that we can fuse the three layers into a fused layer.

In addition, the input image will be divided into a $M \times M$ grid (shown in Figure 5) in the YOLOv2 network, which makes the candidate bounding boxes have the equal density distribution in the direction of X axis and Y axis. Normally, however, the distribution of pedestrians in the X axis is more intensive, while the distribution is sparse on the Y axis such as Figure 4. This splitting technique will lead to a high miss rate of the original network. In view of the above analysis, we add a reorganization layer at the end of the model, equivalent to splitting the input image into $M \times N$ ($M>N$, shown in Figure 6), in order to increase the density of the direction of the X axis.

2.3. Loss Function. To optimize the whole model, we employ the original joint loss function shown in (5) that is designed for YOLOv2:

$$\begin{aligned} L = {} & \beta_{coord} \sum_{i=0}^{M\times N} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\left(x_i - \widehat{x}_i\right)^2 + \left(y_i - \widehat{y}_i\right)^2 \right] \\ & + \beta_{coord} \sum_{i=0}^{M\times N} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{\widehat{w}_i} \right)^2 \right. \\ & \left. + \left(\sqrt{h_i} - \sqrt{\widehat{h}_i} \right)^2 \right] + \sum_{i=0}^{M\times N} \sum_{j=0}^{B} 1_{ij}^{obj} \left(C_i - \widehat{C}_i\right)^2 \\ & + \beta_{noobj} \sum_{i=0}^{M\times N} \sum_{j=0}^{B} 1_{ij}^{noobj} \left(C_i - \widehat{C}_i\right)^2 + \sum_{i=0}^{M\times N} 1_{i}^{obj} \\ & \cdot \sum_{c\in classes} \left(p_i\left(c\right) - \widehat{p}_i\left(c\right)\right)^2 \end{aligned} \tag{5}$$

where 1*obj i* denotes if object presents to unit *i* and 1*obj ij* means that the *jth* box predictor in unit *i* is in charge of that prediction. The first two terms of the formula are used to predict the bounding boxes of objects. Furthermore, the third item is designed to predict the confidence scores of the bounding boxes and the fourth item is applied to predict the confidence score without an object, while the last is intended for predicting the category each cell belongs to.

2.4. Improvement in the Caltech Pedestrian Dataset

Scale-Aware Structure. The Caltech Pedestrian Dataset is a challenging and a commonly accepted dataset with a large scale span. Normally, the large variance in size will result in a

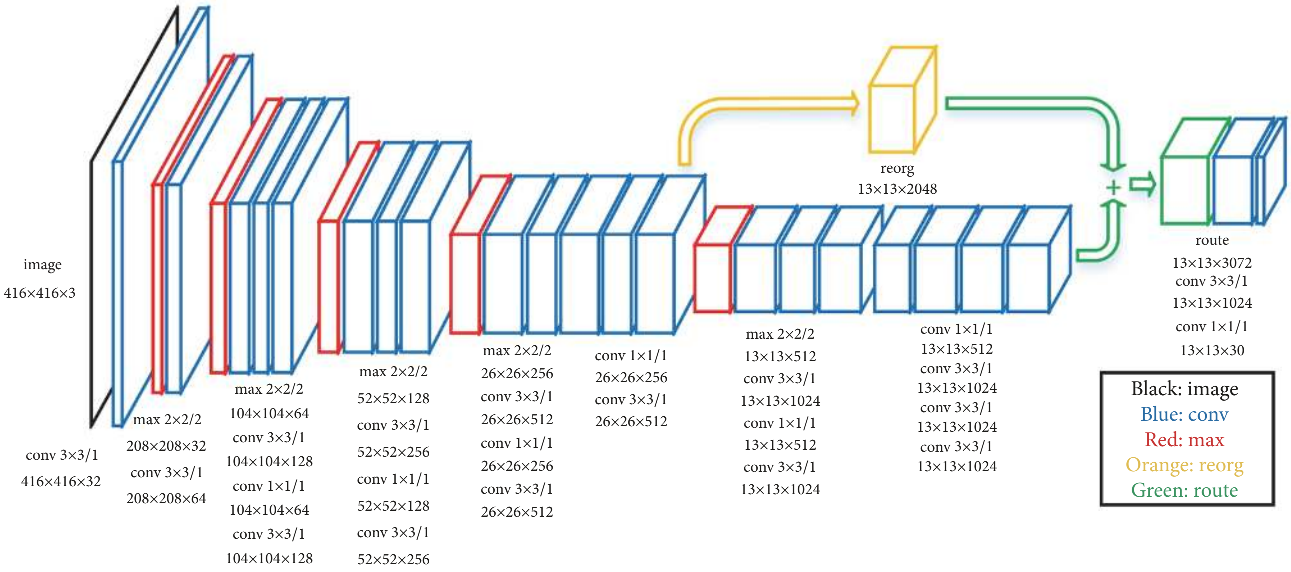


Figure 2: The network architecture of YOLOv2.

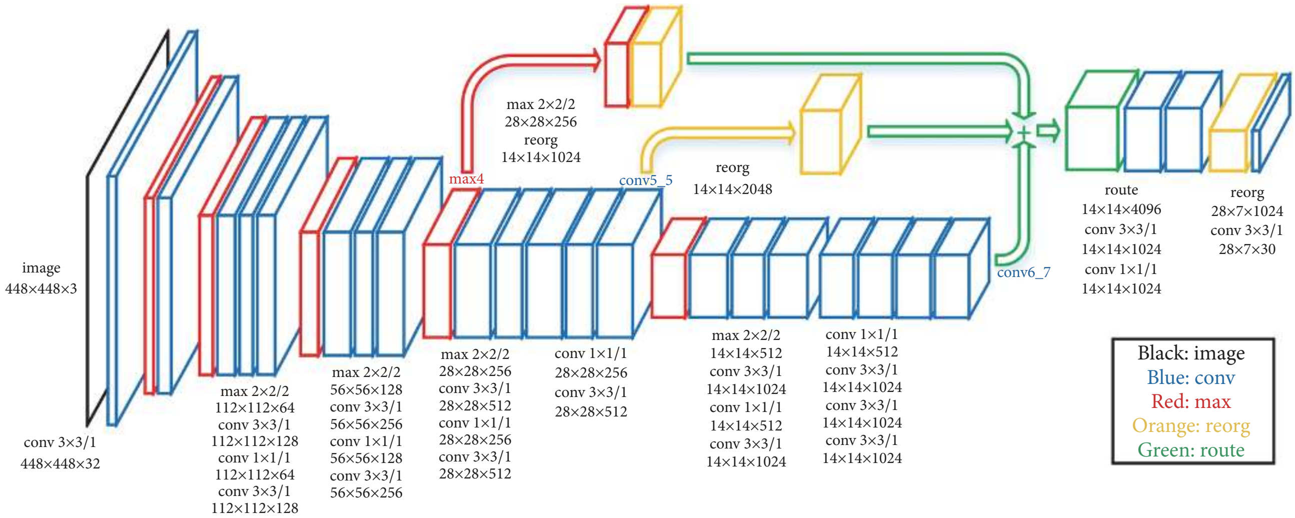


Figure 3: The network framework of our model Y-PD.

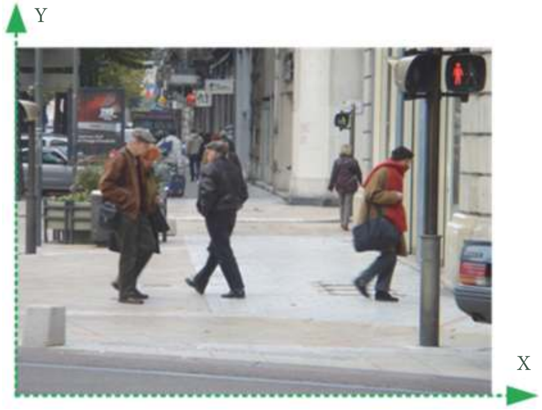


Figure 4: The distribution of pedestrians.

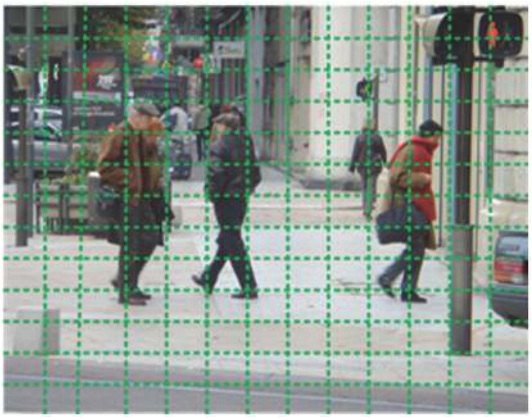


Figure 5: The input image is divided into $M \times M$ grid (13×13).

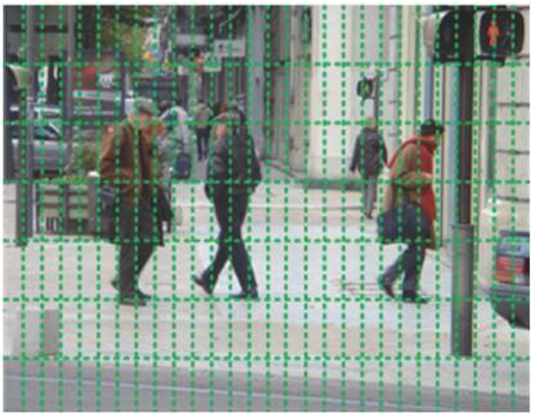


Figure 6: The input image is divided into $M \times N$ grid (28×7).

great intraclass discrepancy, which may hurt the performance of the detection model severely. In order to solve this problem, we employ a scale-aware structure (see Figure 8) inspired by the Scale-Aware Fast R-CNN Network (SA-Faster R-CNN)[23]. We remove several convolution layers from the backbone after *conv6_3*. And then change the downstream layers into two subbranches which are responsible for the large and small size separately. We add the weighted layers to weigh the output feature maps of the two branches pixel by pixel according to the parameter h of each cell. The weights of the two subbranches are w_l and w_h.

$$w_{\mathrm{l}} = \frac{1}{2}\exp\left(-\frac{h-h'}{h+h'}\right) \tag{6}$$

$$w_h = 1 - w_l \tag{7}$$

The final predicted confidence scores $\boldsymbol{s}_p$ and bounding-box regression offset $\boldsymbol{t}_p$ can be computed as follows:

$$s_p = w_l \times s_l + w_h \times s_h, \tag{8}$$

$$t_p = w_l \times t_l + w_h \times t_h, \tag{9}$$

where h' is the meant height of pedestrians on Caltech dataset. $\boldsymbol{s}_s$ and $\boldsymbol{s}_l$ denote the output confidence score of large-size and small-size subbranches, respectively. $\boldsymbol{t}_s$ and $\boldsymbol{t}_l$ denote the output bounding-box regression offsets of large-size and small-size subbranches, respectively.

Weak Semantic Segmentation. Semantic segmentation is a pixel-wise classification technique. We fuse weak semantic segmentation networks into our model as a strong supervision making the most of semantic information of input image and making the feature extraction of shared convolution layers concentrate more on pedestrians, which like illuminating pedestrians. The fused weak semantic segmentation networks constitute only a single convolution layer being attached to conv6_3 for impacting shared convolution layers as far as possible. For optimizing fused weak semantic segmentation networks, we need minimize the loss function, for every location j:

$$L_{seg} = \sum_j L_s\left(\boldsymbol{S}_j, \boldsymbol{S}_j^{\,*}\right) \tag{10}$$

where L_s is a softmax logistic loss, $\boldsymbol{S}_j^{\,*}$ is ground-truth semantic label for location j, and $\boldsymbol{S}_j$ is output of network for location j. The joint loss function is as follows:

$$Loss = L + \lambda L_{seg} \tag{11}$$

where we set $\lambda = 0.2$ by default. Furthermore, because of the lack of semantic segmentation labels in Caltech Pedestrian Dataset, we should make weak training labels to train fused semantic segmentation networks. We utilize bounding boxes of pedestrians to make weak training labels, the pixels inside the bounding box are considered to be pedestrian, and the pixels outside the bounding box are deemed to be scene (see Figure 7).

3. Experimental Evaluation and Analysis

We conduct an extensive experimental campaign to evaluate the performance of our detection model. All times reported are for implementation in a single CPU core (4.0-4.2GHz) of an Intel Core i7 6700k server with 8GB of RAM. A NVIDIA GTX1080Ti GPU is used for CNN computations.

3.1. Dataset

INRIA Pedestrian Dataset. To perform the following experiments, we recourse to the INRIA Pedestrian Dataset, a commonly accepted, multiscales dataset with a certain challenge which is often used to evaluate the performance of the pedestrian detection techniques. The INRIA Pedestrian Dataset is created in the research work [10] for detecting the erect pedestrian in images and videos. It is subdivided into two patterns: (1) raw images with the appropriate annotations and (2) positive images normalized into 64x128 pixel with the raw negative images. We employ the train set and the test set to train and validate our models, respectively, which are contained in the raw images with the appropriate annotations. In this dataset, only the upright persons whose height are greater than 100 are signed in per image. However, the annotation may be incorrect. Sometimes the part of the bounding box labeled can be inside or outside the object, whose influence can be ignored.

The INRIA Pedestrian Dataset contains a train set and a test set. The train set has 614 positive images, with 1237 pedestrians. While the test set has 228 positive images, with 589 pedestrians. Images in dataset have the complex background with an obvious light change. The pedestrians, with different degrees of occlusion, wearing different costumes, have many kinds of scales and changing postures.

Caltech Pedestrian Dataset. The Caltech Pedestrian Dataset consists of a set of video sequences of 640×480 size taken from a vehicle driving in the urban environments. The dataset includes some train (set00-set05) and

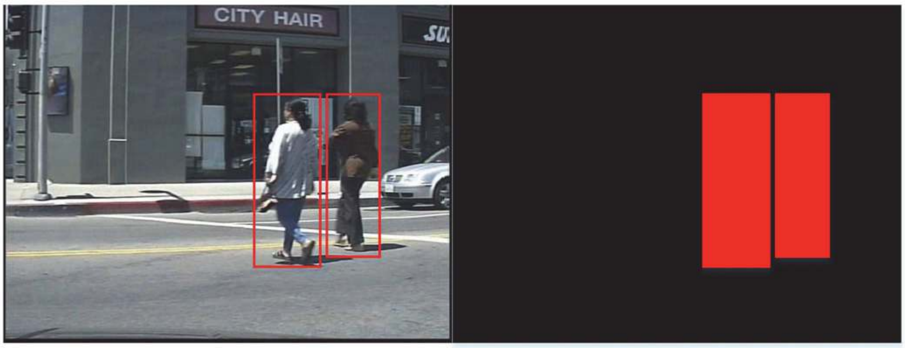


Figure 7: The weak semantic segmentation labels.

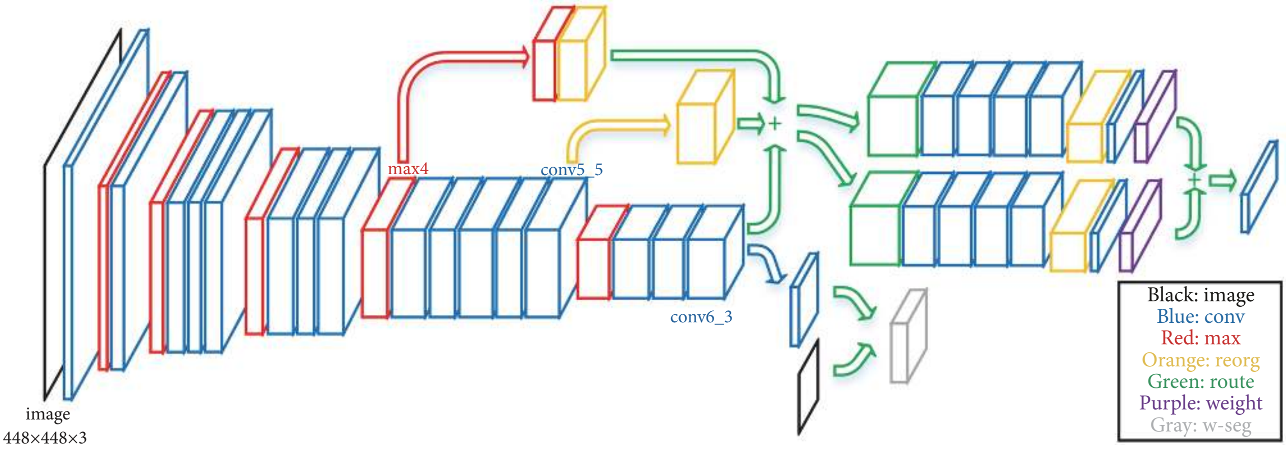


Figure 8: The network framework of our model Y-PD+SSW.

test (set06-set10) subsets. There are about 350000 bounding boxes in 250000 frames with 2300 unique pedestrians annotated. According to the condition of pedestrian, a single object will be assigned one of the four labels, including "person" (~1900), "people" (~300), and "person?" (~110) (only "person" and "people" will be used in our experiment).

3.2. Evaluation Metrics. We recourse to the evaluation metrics defined by the Caltech pedestrian detection evaluation protocol, which is developed by Dollar et al. [24]. Particularly, the performance of a method is assessed in the light of the trade-off between the number of false positives per image (*FPPI*) and the miss rate (*MR*). To make it easier for readers to understand, we will make a brief description of such metrics. First, a ground truth is deemed to match a detected bounding box, provided by pedestrian detection algorithm, if their intersection over union (*IOU*) is greater than 50%. A ground truth will be deemed as a False Negative (*FN*) or a miss if it does not have a match. On the contrary, if a detected box fails to match the ground truth, it will be regarded as a False Positive (*FP*). Then the average number of proposals per image detected as a pedestrian erroneously is regarded as the average number of false positives per image (*FPPI*). And the miss rate (*MR*) donates the ratio between the number of False Negatives and the total number N of positive samples as shown in

$$MR = \frac{FP}{N} \tag{12}$$

Occasionally, we might replace *MR* with *Recall* as shown in

$$Recall = 1 - MR \tag{13}$$

Typically, the miss rate value at 0.1 *FPPI* we pay special attention to has been regarded as a reasonable working condition for an available system in practice.

3.3. Experimental Process and Results

Pretraining Y-PD. We take advantage of the pretraining, which means that the weights of the model are initialized from the weights trained in ImageNet dataset. This technique is one of the most useful measures for improving the performance of deep models, because the number of parameters is generally far greater than the data collected for training. And it makes the algorithm have a faster convergence rate or utilize less available data to obtain the great results. We compare the model without pretraining and observe

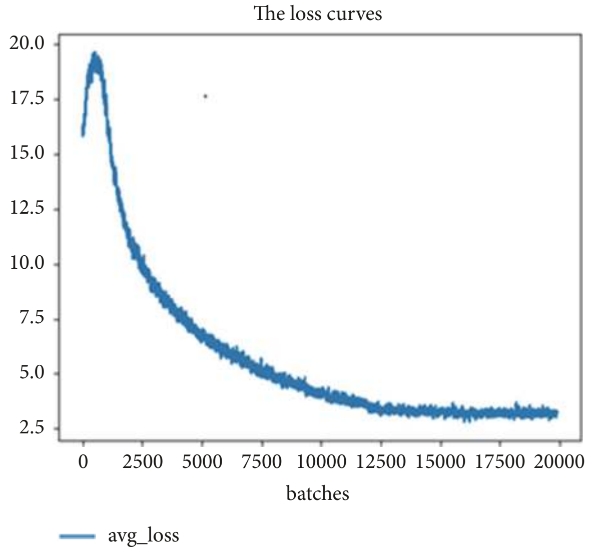


Figure 9: The loss curve without pretraining.

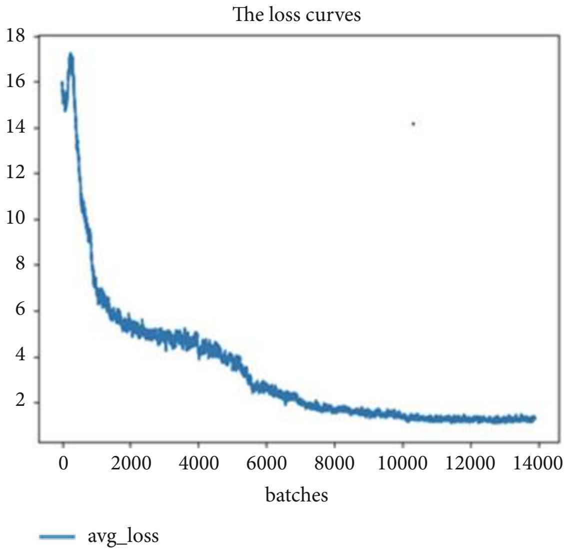


Figure 10: The loss curve with pretraining.

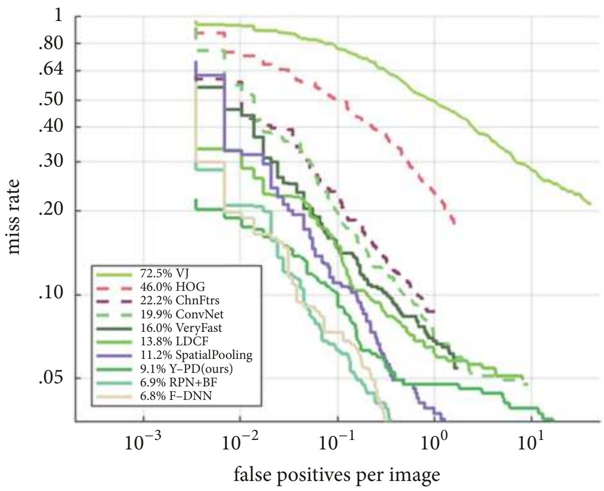


Figure 11: The comparison of methods in the INRIA Pedestrian Dataset (reasonable).

Table 1: The effect of pretraining.

Model	AP
Y-PD with pretraining	90.9%
Y-PD without pretraining	84.5%

the improvement in the accuracy in Table 1, that is, 6.4%. Meanwhile, we can see from the training loss curves in Figures 9 and 10 that the model trained with pretraining converges faster and has a smaller ultimate loss than the one without pretraining.

Compared with the Baseline in the INRIA Pedestrian Dataset. We test all the mends we have taken and observe the improvements compared with the baseline YOLOv2. Check Table 2 for the details, where ChD donates a change in the distribution in the direction of X axis and Y axis and AdL donates an added pass-through layer. When we change the distribution only, the average accuracy can increase 2.8%. If we add pass-through layer only, the average accuracy refers to an increase of 0.7%. While combining the above two measures, the average accuracy is able to increase 3.3%.

Compared with the State-of-the-Art Algorithms in the INRIA Pedestrian Dataset. To establish the performance level of our model, we select several typical algorithms and advanced algorithms to compare with the INRIA test data for pedestrians. The MR-FPPI curves are shown in Figure 11. Furthermore, to embody the advantages of our model, we present Table 3 that shows the average miss rate and the speed of detection for some of the above methods. The speed of detection may not be very accurate for the reason of the limited conditions, but the gap is not too big. Although the

Table 2: Results in INRIA validation set.

Model	FPS	AP	Improvement
YOLOv2(baseline)	84	87.6%	-
YOLOv2+ChD	75	90.4%	+2.8%
YOLOv2+AdL	73	88.3%	+0.7%
YOLOv2+ChD+AdL(Y-PD)	73	90.9%	+3.3%

Table 3: A comparison of speed and average miss rate among methods.

Model	FPS (on GTX1080ti)	Avg. miss rate
VJ	<1	72.5%
HOG	<1	46.0%
VeryFast	>100	16.0%
SpatialPooling	<1	11.2%
Y-PD (ours)	73	9.1%
RPN+BF	~4	6.9%
F-DNN	~6	6.8%

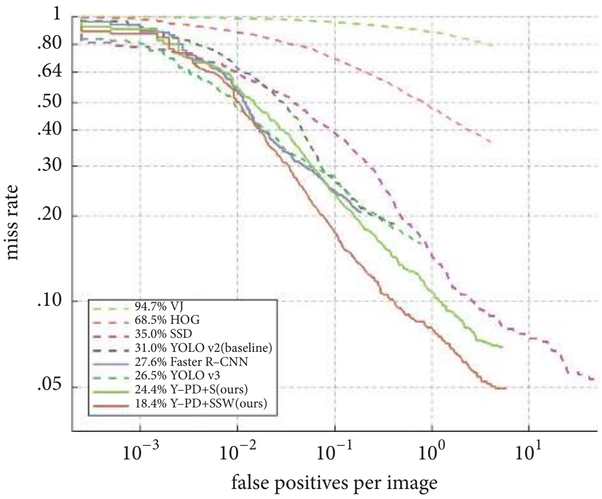


Figure 12: Comparisons of the methods in the Caltech Pedestrian Dataset (reasonable).

precision of our model is not as high as RPN+BF and F-DNN, whose gap is only 2.2% and 2.3%, respectively, the speed of the detection of ours is dozens of times as good as theirs. Obviously, our model is able to achieve a better trade-off between speed and accuracy in INRIA test data for pedestrians.

Compared with the state-of-the-art general detection algorithms in the Caltech Pedestrian Dataset. From Figure 12, we can realize that our model Y-PD+S and Y-PD+SSW have better detection performance compared with YOLO v2, Faster R-CNN, and YOLO v3 when tested in the Caltech Pedestrian Dataset (see Table 4). And the model Y-PD+SSW that employs a scale-aware structure increases by 6% compared with Y-PD+S.

Table 4: A comparison of average miss rate among general methods.

Model	Avg. miss rate
YOLO v2 (baseline)	31.0%
Faster R-CNN	27.6%
YOLO v3	26.5%
Y-PD+S(ours)	24.4%
Y-PD+SSW(ours)	18.4%

4. Summary

In this paper, we present a model named Y-PD for the pedestrians detection based on YOLOv2. The architecture of Y-PD covers the characteristics of pedestrian distribution and takes full advantage of low-level and high-level feature maps. The experiment result shows it can achieve a good trade-off between speed and accuracy in the INRIA test data for the pedestrians. Furthermore, the model Y-PD+SSW employs a scale-aware structure based on Y-PD and fuses the weak semantic segmentation networks and make a great progress in the Caltech dataset. However, because of the diversity of size, resolution and so on, there is still a big gap between our model and the state-of-art pedestrian methods. So our future task will mainly work on designing of the better model of the Caltech dataset for pedestrians.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request, and you can also download the relevant dataset through some links provided in Supplementary Materials.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (61561042), the Talent Introduction Foundation of Northwest Minzu University.

Supplementary Materials

(1) The INRIA Dataset. INRIA dataset was collected as part of research work on detection of upright people in images and video. The research is described in detail in CVPR 2005 paper Histograms of Oriented Gradients for Human Detection and their PhD thesis. The dataset is divided into two formats: (a) original images with corresponding annotation files and (b) positive images in normalized 64x128 pixel format (as used in the CVPR paper) with original negative images. The data set contains images from several different sources: Images from GRAZ 01 dataset, though annotation files are completely new and images from personal digital image collections taken over a long time period. Usually the original positive images were of very high resolution (approx. 2592x1944 pixels), so we have cropped these images to highlight persons. Many people are bystanders taken from the backgrounds of these input photos, so ideally there is no particular bias in their pose. Few images are taken from the web using google images. Only upright persons (with person height > 100) are marked in each image. Annotations may not be right; in particular at times portions of annotated bounding boxes may be outside or inside the object. *(Supplementary Materials)*

References

[1] R. Benenson, M. Mathias, T. Tuytelaars, and L. V. Van Gool, "Seeking the strongest rigid detector," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 3666–3673, Portland, Ore, USA, June 2013.

[2] Y. Ding and J. Xiao, "Contextual boost for pedestrian detection," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2895–2902, USA, June 2012.

[3] L. V. Gool, "Pedestrian detection at 100 frames per second," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2903–2910, 2012.

[4] S. Zhang, C. Bauckhage, and A. B. Cremers, "Informed haar-like features improve pedestrian detection," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 947–954, IEEE, Columbus, OH, USA, June 2014.

[5] P. Luo, Y. Tian, X. Wang, and X. Tang, "Switchable Deep Network for Pedestrian Detection," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 899–905, IEEE, Columbus, OH, USA, June 2014.

[6] S. Paisitkriangkrai, C. Shen, and A. V. D. Hengel, *Strengthening the Effectiveness of Pedestrian Detection with Spatially Pooled Features*, Springer International Publishing, 2014.

[7] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, (CVPR '17)*, pp. 6517–6525, Honolulu, Hawaii, USA, 2017.

[8] R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Ten years of pedestrian detection, what have we learned?" in *Proceedings of the European Conference on Computer Vision (ECCV '14)*, vol. 8926 of *Lecture Notes in Computer Science*, pp. 613–627, 2014.

[9] A. D. Costea and S. Nedevschi, "Word channel based multiscale pedestrian detection without image resizing and using only one classifier," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition, (CVPR 2014)*, pp. 2393–2400, USA, June 2014.

[10] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 886–893, San Diego, CA, USA, 2005.

[11] R. Appel and W. Kienzle, "Crosstalk cascades for frame-rate pedestrian detection," in *Proceedings of the European Conference on Computer Vision (ECCV '12)*, pp. 645–659, 2013.

[12] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li, "Robust multi-resolution pedestrian detection in traffic scenes," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3033–3040, 2013.

[13] W. Ouyang and X. Wang, "Joint Deep Learning for Pedestrian Detection," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, pp. 2056–2063, IEEE, Sydney, NSW, Australia, 2014.

[14] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west," in *Proceedings of the British Machine Vision Conference*, pp. 1–11, BMVA Press, 2010.

[15] X. Zeng, W. Ouyang, and X. Wang, "Multi-stage contextual deep learning for pedestrian detection," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, pp. 121–128, IEEE, Sydney, NSW, Australia, 2013.

[16] C. Wojek, S. Walk, and B. Schiele, "Multi-cue onboard pedestrian detection," in *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 794–801, 2009.

[17] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?" in *Proceedings of the European Conference on Computer Vision*, pp. 443–457, 2016.

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

[19] X. Du, M. El-Khamy, J. Lee, and L. S. Davis, "Fused DNN: a deep neural network fusion approach to fast and robust pedestrian detection," in *Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 953–961, 2016.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, IEEE, Las Vegas, NV, USA, 2016.

[21] T. Kong, A. Yao, Y. Chen, and F. Sun, "HyperNet: towards accurate region proposal generation and joint object detection," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 845–853, IEEE, Las Vegas, NV, USA, 2016.

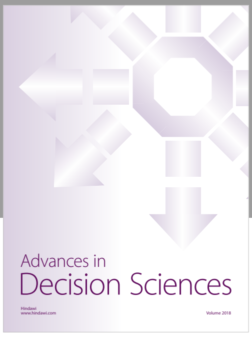
[22] S. Xie and Z. Tu, "Holistically-Nested Edge Detection," in *Proceedings of the 2015 IEEE International Conference on Computer*

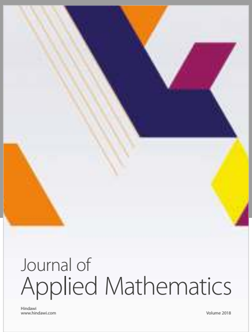
Vision (ICCV), pp. 1395–1403, IEEE, Santiago, Chile, December 2015.

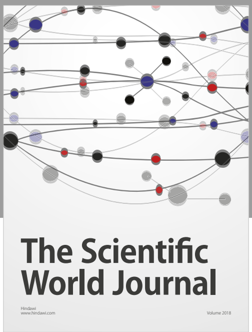
[23] J. Li, X. Liang, S. M. Shen, T. Xu, J. Feng, and S. Yan, “Scale-aware Fast R-CNN for pedestrian detection,” *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 985–996, 2015.

[24] B. Schiele and P. Perona, “Pedestrian detection: an evaluation of the state of the art,” *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.

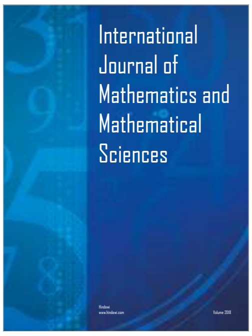
Advances in
Operations Research


Advances in
Decision Sciences

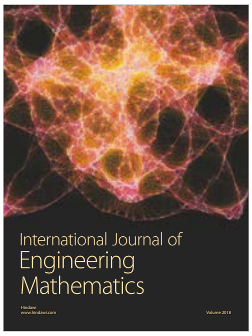

Journal of
Applied Mathematics


The Scientific
World Journal

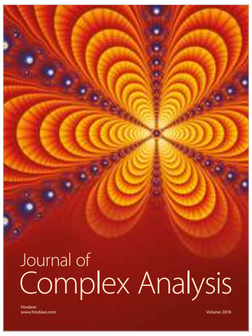
Journal of
Probability and Statistics

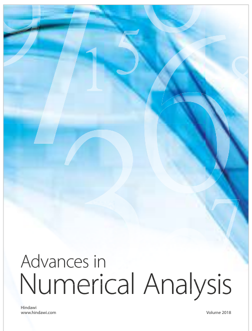

International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Optimization

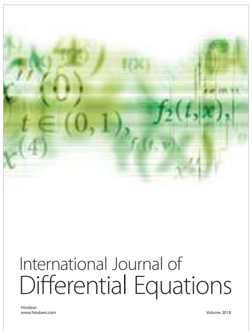

International Journal of
Engineering
Mathematics

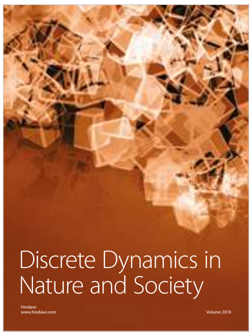
International Journal of
Analysis

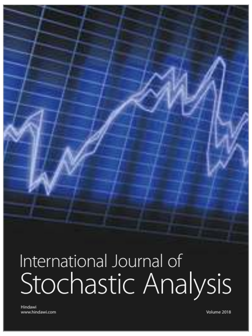

Journal of
Complex Analysis


Advances in
Numerical Analysis

Mathematical Problems
in Engineering

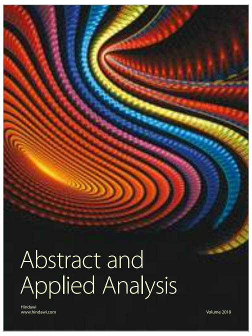

International Journal of
Differential Equations

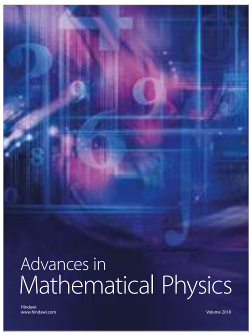

Discrete Dynamics in
Nature and Society


International Journal of
Stochastic Analysis

Journal of
Mathematics

Journal of
Function Spaces


Abstract and
Applied Analysis


Advances in
Mathematical Physics