A scale-aware YOLO model for pedestrian detection

Xingyi Yang[1], Yong Wang[2], and Robert Laganière[3]

[1] Department of Electrical and Computer Engineering, UC San Diego
x3yang@ucsd.edu
[2] School of Aeronautics and Astronautics, Sun Yat-Sen University
wangyong5@mail.sysu.edu.cn
[3] School of Electrical Engineering and Computer Science, University of Ottawa
laganier@eecs.uottawa.ca

Abstract. Pedestrian detection is considered one of the most challenging problems in computer vision, as it involves the combination of classification and localization within a scene. Recently, convolutional neural networks (CNNs) have been demonstrated to achieve superior detection results compared to traditional approaches. Although YOLOv3 (an improved You Only Look Once model) is proposed as one of state-of-the-art methods in CNN-based object detection, it remains very challenging to leverage this method for real-time pedestrian detection. In this paper, we propose a new framework called SA YOLOv3, a scale-aware You Only Look Once framework which improves YOLOv3 in improving pedestrian detection of small scale pedestrian instances in a real-time manner. Our network introduces two sub-networks which detect pedestrians of different scales. Outputs from the sub-networks are then combined to generate robust detection results. Experimental results show that the proposed SA YOLOv3 framework outperforms the results of YOLOv3 on public datasets and run at an average of 11 fps on a GPU.

Keywords: pedestrian detection· YOLO· CNN

1 Introduction

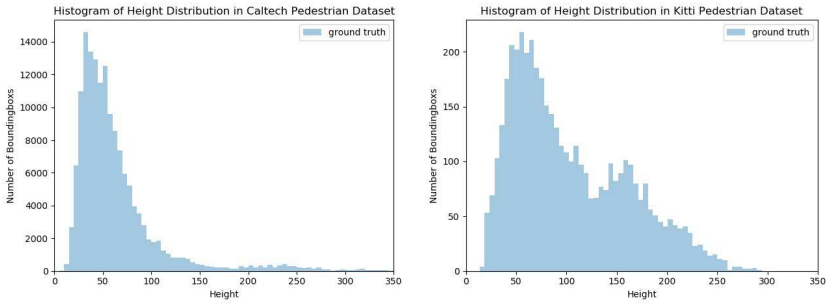
Pedestrian detection is one of the most challenging problems in the field of computer vision. The goal of pedestrian detection is to localize different pedestrians in a scene and assign bounding boxes. It has been the object of many studies in the computer vision community as an important component in many applications including intelligent vehicles, person re-identification and robotics, to name a few.

Traditional methods [1, 2] use a sliding window approach and a classifier is employed to determine the presence of a pedestrian. However, this type of approach has a high detection error rate. Recently, convolutional neural networks (CNNs) have shown significant performance in a range of different applications, with pedestrian detection being one of the key areas where CNNs clearly outperforms traditional approaches [3, 4, 5, 6, 7]. For example, in [6], an end-to-end

CNN architecture is employed to generate pedestrian bounding boxes via multiple layers in an image, and a classifier performs classification on bounding boxes. Although these two-stage methods are able to produce high accuracy, the whole procedure is computationally expensive.

To speed up detection procedure, a framework called You Only Look Once (YOLO) [8] is proposed which formulate the detection problem as a single regression problem, where bounding box position and class probabilities are solved at the same time. YOLO has demonstrated its ability to provide appealing speed advantages compared to two-stage methods, but it has significantly higher location error than these methods. Recently, YOLOv3 [9] was proposed with the objective of further reducing detection errors. A feature pyramid to improve small object detection is used.

Small pedestrian detection is one of the fundamental problem in pedestrian detection. Existing approaches handle the scale-variance problem mainly from two aspects. First, a multi-scale scheme is employed on pedestrians of various sizes [10, 11]. Second, data augmentation is utilized to improve scale-invariance [12, 13]. However, it is difficult to integrate object features of different scale within a single model as the intra-class variance of large-size and small-size objects is large. In [14], a scale-aware architecture network is proposed to exploit the different characteristics of pedestrians at various scales. However, this is a two stage approach that needs ACF [15] to provide candidate bounding boxes first, which reduces its detection efficiency and accuracy.

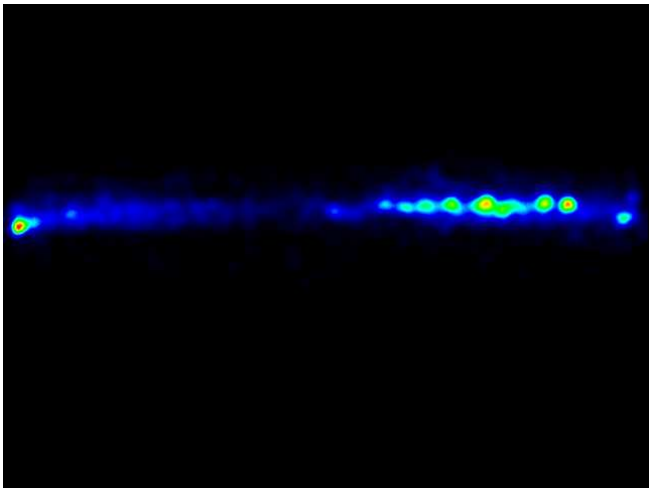


(a) Caltech Pedestrian Dataset (b) Kitti Pedestrian Dataset

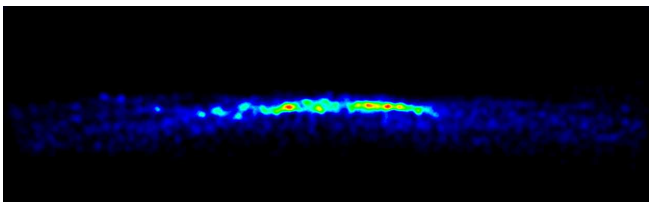
Fig. 1. Histogram of object height (in pixel) distribution for two pedestrian deetction dataset

To address this problem, we further investigate some statistics properties of pedestrian detection. First, small objects dominate pedestrian datasets. Figure 1 shows the histograms of object height on two pedestrian detection dataset. Half of bounding boxes in Caltech dataset [31] and one third from the Kitti dataset [3] have a height less than 50 pixels. Second, small pedestrians tend to appear at the center of the images. Figure 2 visualizes the location heatmap for small

pedestrian (height less than 50). Most bounding boxes center around the middle line of the whole images. Findings above give us two valuable inspirations: 1. We need to design a separated component for small object detection due to its large quantity. 2. We need to specially focus on central image to detect tiny pedestrians.



(a) Caltech Pedestrian Dataset



(b) Kitti Pedestrian Dataset

Fig. 2. Pedestrian location heatmap with height less than 50 pixels for two pedestrian deetction dataset. Brighter colors indicate more objects gathered abound that location.

Motivated by the above observations, we develop a novel scale-aware YOLOv3 framework which is built on the YOLOv3 [9]. Different from [14], this framework is an end-to-end architecture and one-stage detector. The proposed scale-aware YOLOv3 integrates a large-size sub-network and a small-size sub-network. Different scales pedestrians are trained with different sub-networks.

The contribution of our work is as follows. First, we propose a novel scale-aware YOLOv3 model for pedestrian detection by integrating a large-size sub-network and a small-size sub-network into a unified architecture. Second, different training strategies are implemented on the two sub-networks. Third, ex-

tensive experiments on challenging pedestrian datasets demonstrate that our method achieve state-of-the-art performance.

The paper is organized as follows. Section 2 reviews the related works. In Section 3, the methodology of our method is described in details. In Section 4, experimental results demonstrating the efficacy of our method are presented. Conclusions are drawn in Section 5.

2 Related works

There are a great number of literatures on pedestrian detection. We mainly focus here on efforts that are closely related to our method.

Hand-crafted features have played a key role in obtaining good performance. The histogram of oriented gradient (HOG) descriptor [1] is one of the most well-known features constructed for pedestrian detection. It has been improved through the introduction of integral channel features (ICF) in [17]. Features such as Haar features, histograms, and local sums are efficiently computed using integral images. This work has been further extended in several ways, e.g., ACF [15].

In recent years, CNN-based approaches [18, 19, 20, 21] have made significant improvements in pedestrian detection. The RCNN developed in [13] combines object proposals with CNN features. This leads to SPPnet [22] which enhances the detection speed of RCNN by computing CNN features once per image. Built on top of RCNN, Fast-RCNN [12] combines single-stage training with multi-task learning of a classifier and a bounding box regressor. Moreover, a region proposal network is developed in Faster-RCNN [23]. It shares entire image CNN features with the detection network to effectively predict object position, leading to a significant speedup for detection.

The RPN+BF model [7] demonstrated that the RPN performs well as a detector while the classifier degrades in performance due to collapsing bins of small-size pedestrians. This problem can be alleviated by using higher resolution features and replacing the classifier with a boosted forest. F-DNN [24] also adopts the Faster R-CNN framework. It fuses multiple classifiers including ResNet [25] and GoogLeNet [26] by using soft-reject and incorporates multiple training datasets.

A convolutional sparse coding based method is employed in [27] to pre-train CNN for pedestrian detection. In [21] pedestrian detection is jointly optimized with other semantic tasks including scene attributes. Complexity-aware cascaded detectors are introduced in [18] by leveraging both CNN and hand-crafted features for trade-off between speed and accuracy.

Multi-layer methods have also been proposed for detecting objects across various sizes. Up-sampling inputs training and testing are used in [6, 22] to improve the scale-invariance of Faster RCNN [23]. SA-FastRCNN [14] proposes two sub-networks based on Fast-RCNN to adaptively detect pedestrians across different scales. Similarly, multiple layers are utilized in MSCNN [6] to match objects of different sizes.

Complementary detectors can be integrated to create a strong multi-scale detector. A single classifier is trained at a fixed resolution [13, 22]. Then the input image is resized to several different scales and the associated features are computed independently.

Various one-stage detector methods have been proposed to better balance the detection accuracy and speed [8, 9, 28, 29]. SSD [28] discretizes the output space of bounding boxes into a set of template boxes over varying aspect ratios and scales. An improved YOLO is proposed in [29] (named YOLOv2) where anchor boxes are employed to predict bounding boxes. In addition, there is no fully-connected layer in YOLOv2. To improve the training model accuracy, k-means clustering is adopted on the training set to automatically select good priors. The Darknet-19 model was developed to make YOLOv2 faster.

This motivates us to consider a simpler grouping of pedestrians into two sizes, large size and small size, which corresponds to near vs. far instances in dataset. The focus of this paper is to achieve a more balanced detection performance for both large and small-size pedestrian images.

3 Our detection algorithm

3.1 Overview of our framework

As shown in Figure 3, given an input image, the SA YOLOv3 first divides the image into two parts. The whole image is for large scale pedestrian detection. And the center part is for small scale pedestrian detection. These two parts are first passed through the shared convolutional layers to extract corresponding feature maps. Then they are separately sent to two sub-networks. Different confidence scores and bounding boxes are assigned which are then combined to generate the final detection results using NMS.

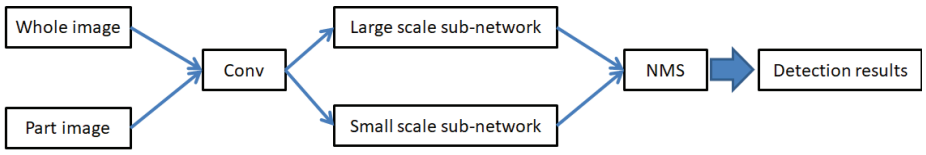


Fig. 3. Illustration of our SA YOLOv3. A large-scale and a small-scale sub-network are learned specifically to detect pedestrians with different scales. The final result is obtained by fusing the outputs of the two sub-networks.

3.2 YOLOv3

YOLOv3 improves YOLO by proposing several extensions. The first is class prediction. Binary cross-entropy loss is utilized during training. The second is

the introduction of a new network, Darknet-53, as illustrated in Figure 4. This network is an improved version of Darknet-19 and more efficient than ResNet-101 and ResNet-152. The third is multiple scale prediction. Three layers are employed to predict bounding boxes using a feature pyramid similar to [30]. These multiple features can provide more meaningful semantic information from the deeper layers and finer grained information from the earlier feature maps.

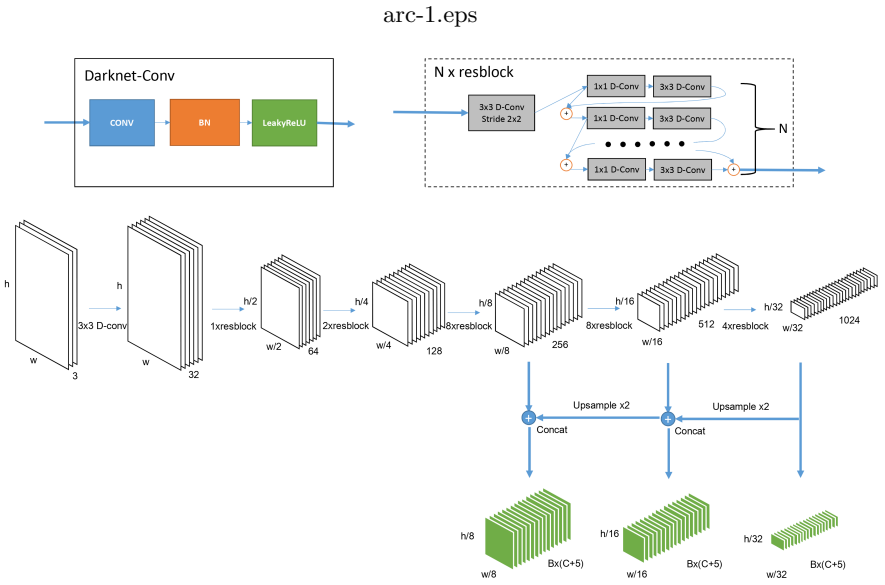


Fig. 4. The Darknet-53 model.

3.3 Architecture of SA YOLOv3

Figure 5 illustrates the architecture of SA YOLOv3. The input image is first divided into three parts according to the scene geometry in which small scale pedestrians usually appear in the middle of the images. The whole image and center part are sent into the network. Several convolutional layers are used to extract feature maps. Then the proposed network branches into two sub-networks, which are learned specifically to detect large-scale and small-scale pedestrians respectively.

Each sub-network takes as input the feature maps generated from the previous convolutional layers. These feature maps are further extracted through a sequence of convolutional layers to generate feature specialized for a specific scale. Feature maps are pooled into a fixed-length feature vector which is fed into a sequence of fully connected layers.

Each sub-network follows two output layers which produce two output vectors per instance proposal. Specifically, one layer outputs classification scores, the other one regresses the bounding box coordinate. Finally, the outputs from the two sub-networks are fused via NMS.

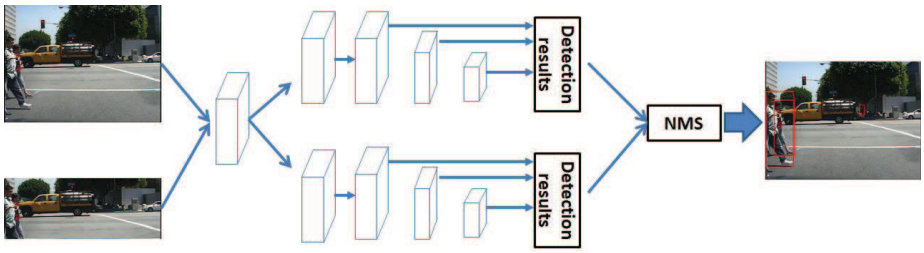


Fig. 5. The architecture of our SA YOLOv3. The features of the entire input image are first extracted by a set of convolutional layers, and then fed into two sub-networks. A sequence of convolutional layers is utilized to further extract scale-specific features in each sub-network. Next, the produced feature maps are pooled into a fixed-length feature vector via a RoI pooling layer. Then several fully connected layers generate scale-specific detection results: one outputs classification scores and the other outputs refined bounding box coordinate for each pedestrian. Finally, the outputs of the two sub-networks are fused by applying NMS.

4 Experiment

We evaluate our method on the popular Caltech dataset [31]. Comprehensive analysis and ablation experiments are carried out using the Caltech dataset. In addtition, to test generalization of our model, pedestrian detection is also carried out on the KITTI dataset [3].

4.1 Implementation details

The Caltech dataset [31] consists of 350K pedestrian bounding box annotations across 10 hours of urban driving. The log average miss rate against a false positive per image (FPPI) range of [10 2; 10 0] is utilized for evaluating performance. A minimum intersection over union (IoU) threshold of 0.5 is required for a detected box to match with a ground truth box. For training, we sample every 5 frames from the standard training set, which contains 10734 training images. We evaluate on the standard 4024 images in the testing set. In our experiments, six subsets are considered to demonstrate the performance on occlusion and small size issues: reasonable, all-scale, far-scale, large, medium-scale, heavy occluded. In the reasonable subset, pedestrians are over 50 pixels. In the all-scale subset,

pedestrians are over 20 pixels. In the far-scale subset, pedestrians are between 20 to 30 pixels. In the large subset, pedestrians are over 100 pixels. In the medium-scale subset, pedestrians are between 30 to 80 pixels. In the heavy occluded subset, pedestrians are 36% to 80% occluded.

The input size in KITTI dataset is 375×1242. The detection results are uploaded to KITTI website and the results are evaluated by mean Average Precision (mAP) which is the area under the Precision-Recall curve. There are three difficulty levels: easy, moderate and hard.

The heights of pedestrians used to train the small sub-network and large sub-network are below 50 pixels and larger than 30 pixels respectively.

4.2 Quantitative comparison

Caltech dataset For comparison, we enlist here a group of nine algorithms, including HOG [1], VJ [2], SSD-resnet50 [28], SA-FastRCNN [14], YOLOv3 [9], SDS-RCNN [32], MS-CNN [6], AdaptFastRCNN [33], F-DNN+SS [24]. Evaluation results are measured in terms of the log-average miss rate for pedestrian instances of the six situations.

Figure.6 (a) displays the quantitative results of reasonable. Our approach (16%) outperforms YOLOv3 [9] (20%) and reduce the FPPI by an average of 20%. Meanwhile, our method also outperforms the one stage method SSD-resnet50 [28] (20%). SDS-RCNN [32] and F-DNN+SS [24] achieve the lowest and second lowest log-average miss rate as segmentation methods are employed. AdaptFasterRCNN [33], SA-fastRCNN [14] and MS-CNN [6] performs slightly better than ours. These methods are two-stage RCNN methods which are slower than our method.

Further, for all (b), our approach achieves the second lowest missing rate of 53%, which results in substantially better performance than the existing results, e.g., 60% of AdaptFasterRCNN [33] and 61% of MS-CNN [6].

Far-scale (c). Our approach significantly outperforms all compared methods and achieves the lowest log-average miss rate of 72%, which exceeds the results 77% of F-DNN+SS [24]. As the amount of hard-to-detect far-scale instances dominates the overall pedestrian population of Caltech benchmark, our framework contributes an effective solution.

Medium-scale (e). Our method outperforms the other methods except F-DNN+SS [24] in a trend similar to that of (b).

Heavy occluded (f). Our method achieves the best results in FPPI (48%). [31] showed that nearly 70% of the pedestrians are occluded in realistic videos and the detection results degrades rapidly with heavy occlusion. Our method is able to locate pedestrians only partially visible.

KITTI dataset Table 1 shows the comparison of our method on pedestrian detection with other compared methods. Our approach outperforms the vanilla YOLOv3 method and improves 5.5% on hard condition. In addition, our detection time is comparable with YOLOv3and much faster than the other methods.

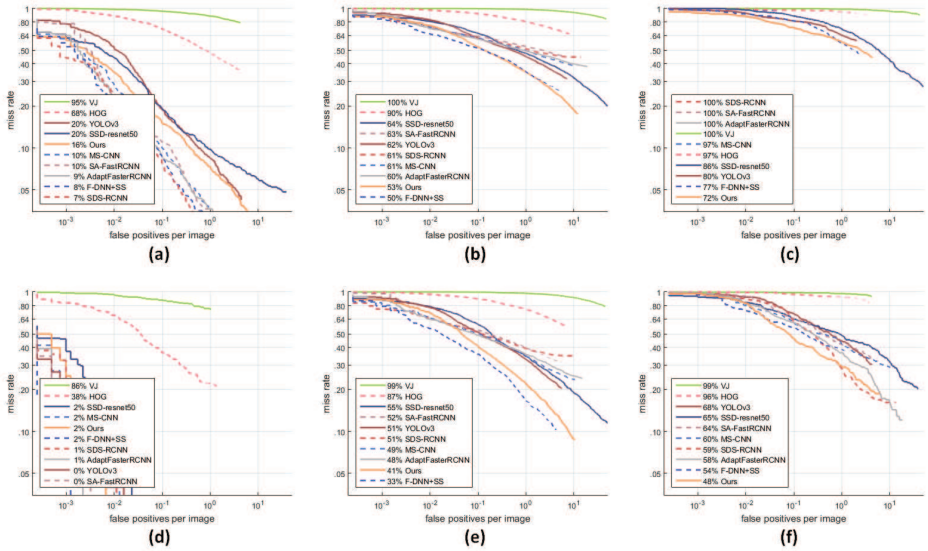


Fig. 6. Quantitative comparison results on the Caltech benchmark. (a) Reasonable. (b) All-scale. (c) Far-scale. (d) Large. (e) Medium-scale. (f) Heavy occluded.

Efficiency is one of the advantages of our framework. Our method takes 11 frames per second (fps) with an input image of size 375*1242 on a single Nvidia Titan Xp GPU. Compared to the RPN+BF [7], our approach executes six times faster (fifth column of Table 1). Our method is slower than YOLOv3 as the structure of our network architecture while our method is more elaborated.

Table 1. The comparison of our method on pedestrian detection with other compared methods on the KITTI dataset.

Method	AP on Easy	AP on Moderate	AP on Hard	Times (s)
RPN+BF [7]	75.58%	61.29%	56.08%	0.6
Faster R-CNN [23]	78.35%	65.91%	61.19%	2
YOLOv3 [9]	77.74%	64.3%	59.00%	0.043
Ours	77.88%	66.69%	62.22%	0.09

4.3 Qualitative comparison

Figure 7 presents examples of detection results of YOLOv3 [9], SA FastRCNN [14] and our method on Caltech dataset to further demonstrate the superiority of our method in detecting small-scale instances. The three columns show the detection results by YOLOv3 [9] and SA FastRCNN [14] and our SA YOLOv3.

The red rectangles represent ground-truth bounding boxes of pedestrians, and the detected instances by our SA YOLOv3 and the two baselines are annotated in green rectangles. One can observe that our method can successfully detect most of the small-scale pedestrian instances that YOLOv3 and SA FastRCNN have missed. It also shows that our method is robust to heavy occlusion of pedestrians, illumination and large background clutters.

5 Conclusion

In this paper, we introduced SA YOLOv3, a new framework for the detection of small scale pedestrians. Although a set of state-of-the-art methods has been proposed, they have difficulty in detecting small pedestrians and often can not be used in real-time. Here, we made fully use of geometry property of input image and take advantage of the YOLOv3 framework to produce a network architecture. Our network consists of two sub-networks to handle two different pedestrian scales. Experimental results show that the proposed SA YOLOv3 framework can reduce the FPPI of the original YOLOv3 method by 20% on the reasonable condition of the Caltech dataset and 5.5% on the hard condition of the KITTI dataset.

References

1. N. Dalal and B. Triggs, Histogram of Oriented Gradient for Human Detection, CVPR 2005, San Diego, California.
2. P. Viola and M. Jones, Robust Real-Time Face Detection, IJCV 2004.
3. Andreas Geiger, Philip Lenz, and Raquel Urtasun, Are we ready for autonomous driving? the kitti vision benchmark suite, In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
4. Zhang, Shanshan, Rodrigo Benenson, and Bernt Schiele, Citypersons: A diverse dataset for pedestrian detection, In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, no. 2, p. 3. 2017.
5. Si Wu, Shufeng Wang, Robert Laganire, Cheng Liu, Hau-San Wong, Yong Xu, Exploiting Target Data to Learn Deep Convolutional Networks for Scene-Adapted Human Detection, IEEE Trans. Image Processing 27(3), pp. 1418-1432, 2018.
6. Z. Cai, Q. Fan, R. Feris, and N. Vasconcelos, A Unified Multi-scale Deep Convolutional Neural Network for Fast Object Detection, ECCV 2016, Amsterdam, The Netherland.
7. L. Zhang, L. Lin, X. Liang, K. He, Is Faster R-CNN Doing Well for Pedestrian Detection? ECCV 2016, Amsterdam, the Netherlands.
8. 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.
9. Redmon, Joseph, and Ali Farhadi, Yolov3: An incremental improvement, arXiv preprint arXiv:1804.02767, 2018.
10. Yunchao Gong, Liwei Wang, Ruiqi Guo, and Svetlana Lazebnik, Multiscale orderless pooling of deep convolutional activation features, In ECCV, pages 392-407. 2014.

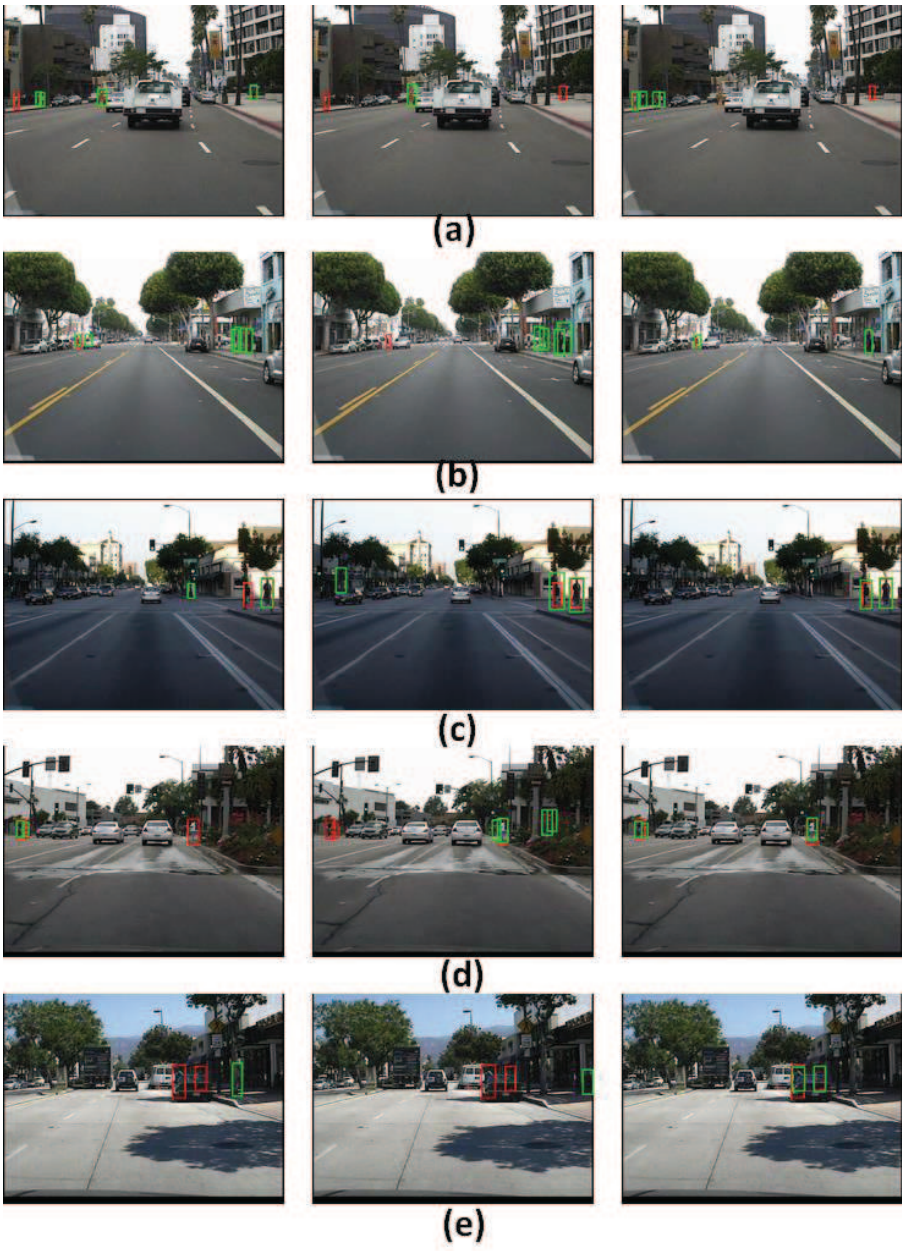


Fig. 7. Visual comparison of YOLOv3, SA Fast RCNN and our detection results on the Caltech benchmark dataset. The three columns sequentially show the detection results by YOLOv3 [9] and SA FastRCNN [14] and our SA YOLOv3. The red rectangles represent ground-truth bounding boxes of pedestrians, and the detected instances by our SA YOLOv3 and the two baselines are annotated in green rectangles.

11. Yichong Xu, Tianjun Xiao, Jiaxing Zhang, Kuiyuan Yang, and Zheng Zhang, Scale-invariant convolutional neural networks, arXiv preprint arXiv:1411.6369, 2014.
12. Girshick, Ross, Fast r-cnn, In Proceedings of the IEEE international conference on computer vision, pp. 1440-1448. 2015.
13. Ross Girshick, Jeff Donahue, Trevor Darrell, and Jagannath Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, In CVPR, pages 580-587, 2014.
14. 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 pp. 985-996, 2018.
15. P. Dollar, R. Appel, S. Belongie, and P. Perona, Fast Feature Pyramids for Object Detection, IEEE TPAMI, 36(8), pp. 1532-45, 2014.
16. Chen, Yuhua, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool, Domain adaptive faster r-cnn for object detection in the wild, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3339-3348. 2018.
17. P. Dollar, Z. Tu, P. Perona, and S. Belongie, Integral channel features, In BMVC, 2009.
18. Z. Cai, M. Saberian, and N. Vasconcelos, Learning complexity-aware cascades for deep pedestrian detection, In ICCV, 2015.
19. J. Hosang, M. Omran, R. Benenson, and B. Schiele, Taking a deeper look at pedestrians, In CVPR, 2015.
20. Y. Tian, P. Luo, X. Wang, and X. Tang, Deep learning strong parts for pedestrian detection, In ICCV, 2015.
21. Y. Tian, P. Luo, X. Wang, and X. Tang, Pedestrian detection aided by deep learning semantic tasks, In CVPR, 2015.
22. K. He, X. Zhang, S. Ren, and J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, In ECCV, 2014.
23. S. Ren, K. He, R. Girshick, and J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, In NIPS, 2015.
24. 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 Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on, pp. 953-961, IEEE, 2017.
25. K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778, 2016.
26. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, Going deeper with convolutions, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-9, 2015.
27. P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun, Pedestrian detection with unsupervised multi-stage feature learning, In CVPR, 2013.
28. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. Berg, SSD: Single shot multibox detector, In ECCV, 2016.
29. Redmon, Joseph, and Ali Farhadi, YOLO9000: better, faster, stronger, arXiv preprint, 2017.
30. T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, Feature pyramid networks for object detection, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2117-2125, 2017.
31. P. Dollar, C. Wojek, B. Schiele, and P. Perona, Pedestrian detection: An evaluation of the state of the art, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34(4), pp. 743-761, April 2012.

32. Brazil, Garrick, Xi Yin, and Xiaoming Liu, Illuminating pedestrians via simultaneous detection segmentation, arXiv preprint arXiv:1706.08564, 2017.