PEDESTRIAN DETECTION WITH DYNAMIC ITERATIVE BOOTSTRAPPING

Chao Pei, Lei Hao, Yuesheng Zhu

Communication and Information Security Lab, Institute of Big Data Technologies,
Shenzhen Graduate School, Peking University

ABSTRACT

Recent years have seen the increasing importance of pedestrian detection, which is a key problem in computer vision. In this paper, we propose a novel pedestrian detection approach based on Faster R-CNN. In order to obtain high-quality candidate regions, relevant adjustments with more precise anchors are made for region proposal network. To resolve the data imbalance issue in the classifier training, we propose a dynamic iterative bootstrapping method where the hard negative examples are automatically selected and the weights of the network are updated iteratively by them to make the training more effective. The square method is used to optimize the multi-task loss in our approach, which can accelerate convergence and reduce sensitivity. Experimental results on different widely used benchmark datasets show that the proposed approach achieves better performance in comparison with other common methods.

Index Terms— Pedestrian detection, Faster R-CNN, region proposal network, bootstrapping.

1. INTRODUCTION

Traditional detectors are trained typically by hand-crafted features. The popular methods include Integral Channel Features (ICF) [1, 2, 3] and Deformable Part based Model (DPM) [4]. The ICF detector was constructed by using multiple types of features (such as LBP [1], SIFT [5], etc.) and a cascade of boosted classifiers. The DPM detector further developed the HOG [6] framework by taking the internal structure of people into account, and achieved state-of-the-art performance.

Recently, the convolutional neural network (CNN) started to attract more and more attention in computer vision, including pedestrian detection. In [7], a model with two convolutional layers was proposed to learn features at all levels in a hierarchy. Benenson et al. [8] reviewed proposal generation algorithms and utilized SCF to reduce the number of candidate regions while Tian et al. [9] utilized ACF [10] in TA-CNN model to obtain high-quality proposals. Their work illustrates the feasibility of Region based CNN(R-CNN) on pedestrian detection. Furthermore, to solve the occlusion issue, Luo et al. [11] proposed the DeepParts framework, which combined the GoogleNet [12] with LDCF [13] and obtained excellent results on standard benchmarks.

However, effective end-to-end CNN-based pedestrian detectors are still rare. The Faster R-CNN [14] framework appears to be a solution. It generates proposals with a fully convolutional Region Proposal Network (RPN) instead of relying on traditional features, and utilizes a CNN classifier to build an end-to-end detect framework. But when directly applied to pedestrian detection, the detector can not achieve the expected result in experiments. We have observed that the quality of proposals still needs to be improved further. The ratio of negative and positive examples for training the classifier are imbalanced, which seriously impacted the performance. Besides, different from multi-category detection, false positives in pedestrian detection are primarily caused by the confusions of hard backgrounds, which should be taken into consideration as well.

In this paper, we propose a novel and effective detector for pedestrian detection by developing the Faster R-CNN framework. The major contributions are threefold. First, we investigate the variety of pedestrians in real scenes and design a pedestrian RPN (PRPN) to get higher quality proposals. Second, to solve the problem of data imbalance in training, we propose the dynamic iterative bootstrapping for the deep network to automatically exploit hard examples and use them to train the detector more effectively. Different from other methods based on bootstrapping algorithm, our approach performs the selecting and training operations at each iteration and we pay more attention to hard negative examples to solve confusions of complex background in particular. Third, based on the impact analysis of factors in multi-task loss, we utilize the square method to optimize the multi-task loss for our approach. It can speed up the convergence of multi-task and make the detector more robust to anomaly data.

2. PEDESTRIAN REGION PROPOSAL NETWORK

RPN was proposed in Faster R-CNN to replace the Selective-Search [15] method for generating candidate regions. It slides a mini-network on feature maps extracted by ConvNets, and predicts k anchors in each position. Despite adjustments of the anchors' spatial locations by bounding-box (bbox) regressor, the regressor could be disturbed seriously that naturally affect the quality of proposals if there are a large number of

inappropriate initial anchors (both ratios and scales). So we propose a pedestrian RPN (PRPN) by tailoring the RPN for pedestrian detection with more precise scales of anchors.

Considering that body proportions of people are similar, we fix the ratio (height to width) as a suitable single value r. On the other hand, to satisfy the requirement of feature pyramids in multi-scale detection, we redefine the anchor width as: $\varphi \cdot \omega^{[\alpha_1,\alpha_2,\cdots,\alpha_n]}$, where φ is baseline scale and $\omega^{[\alpha_1,\alpha_2,\cdots,\alpha_n]}$ is scaling stride.

To generate proposals, the PRPN utilizes the VGG16-net [16] for extracting features and slides a 3×3 convolutional mini-network on the top of the last convolutional layer (*conv5_3* of VGG16-net). Two sibling fully-connected layers (classification and bounding-box regression) receive feature vectors of each sliding area from the mini-network, then predict classes and spatial locations of k anchors. The classification layer outputs $2k$ scores that estimate the probability of background (bg) and foreground (fg) of each proposal, which could be used to build the mini-batch for training classifier.

3. DYNAMIC ITERATIVE BOOTSTRAPPING

In this section, we focus on the training of CNN-based classifier in pedestrian detection. We have found that the issue of data imbalance is serious while the effect of current solutions is not enough, so we propose a dynamic iterative bootstrapping method to handle it and make more efficient training in pedestrian detection.

3.1. Data imbalance and complex background issues

To maintain an end-to-end CNN-based detection framework, we still use a CNN-based classifier behind PRPN as shown in Fig.1. Since the number of proposals is large, we define positive and negative Region of Interests (RoIs) and sample different types of RoIs to build the mini-batch.

Positive RoIs. Positive RoIs are those whose Intersection over Union (IoU) with a ground-truth is greater than a threshold: th_{fg}. We set th_{fg} to 0.5, which is a fairly standard design choice used in Faster R-CNN.

Negative RoIs. Negative RoIs have maximum IoU with ground-truth bbox in the interval $[th_{bg}, th_{fg})$, where the th_{bg} is set to 0.1 in Faster R-CNN. The motivation is that a candidate region which has overlaps with ground-truth is more likely to be considered as a hard negative example. Since we select hard negative examples by bootstrapping, the value is set to 0 (see explanation in sec.3.2).

As defined before, the ratio of positive and negative RoIs can be as high as 1:100. Faster R-CNN attempts to eliminate the imbalance by sampling positive and negative RoIs in each mini-batch and limiting the ratio to 1:3 as much as possible, but it dosen't have the expected results in pedestrian detection. We recorded the actual ratio of each training iteration, and the statistical results showed that the number of iterations

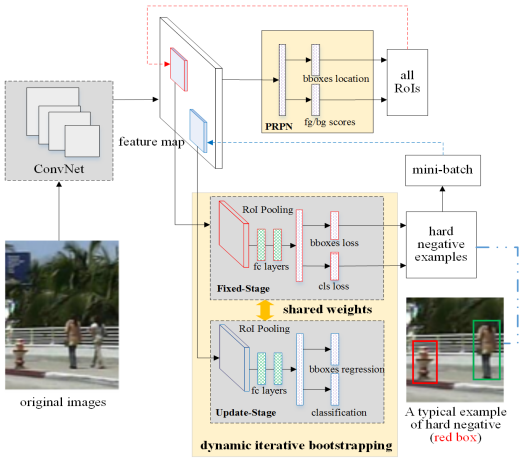


Fig. 1. Architecture of the proposed method

which can really meet the anticipation (1/3) is just about half. The ratio can even reach to 1:30 in some abnormal images.

Besides, it's difficult to select truly useful negative examples from a large number of background examples. Faster R-CNN adopts a strategy of selecting negative RoIs overlaping with ground-truth. However, there are a lot of ambiguous objects with similar features to people in real scenes, which are confusing goals but have no overlap with ground-truth (Fig.2, examples from Caltech dataset [17]). So it is not enough to select hard examples just by increasing th_{bg}. Without useful negative examples, the trained classifier cannot cope with complex conditions.

3.2. Dynamic iterative bootstrapping

To handle the data imbalance issue and harvest more useful hard examples, the dynamic iterative bootstrapping method is proposed by introducing the bootstrapping to deep networks. The bootstrapping has been widely used in traditional classifiers training of object detection research [6, 18]. To use it, the weights of models are fixed at some stages to find hard examples while updated at other stages by these examples. The two stages conduct alternatively to achieve optimization. For deep networks, there are tens of thousands of iterations when the network is trained by Stochastic Gradient Descent (SGD), which makes it difficult to directly utilize bootstrapping. Our approach resolves the problem by dividing each iteration into two stages. The architecture is shown in Fig.1.

Fixed-stage. At the beginning of each iteration, to detect all RoIs with the current network, we fix the weights and set th_{bg} to 0.0 (instead of increasing it) in this stage, then we can obtain the losses of all RoIs. Unlike other variants of bootstrapping used in deep networks that select hard examples only by their losses, we pay more attention to confusions of complex backgrounds and false positives, so our approach selects the hard examples following the rules below: (1) *a negative*

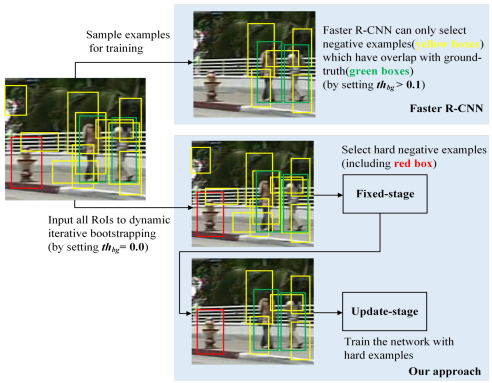
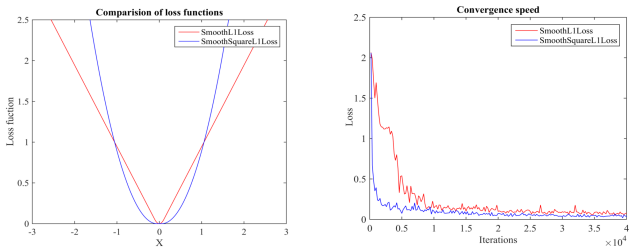


Fig. 2. The comparison of selecting hard negative examples between our approach and Faster R-CNN



(a) Loss function (b) Loss performance

Fig. 3. The comparison between the SmoothSquareL1Loss and SmoothL1Loss.

example is incorrectly classified as positive; (2) *an example (positive or negative) with high loss in the current model*.
To build a useful mini-batch for training, it's worth noting that the mini-batch is still set to 256 and the positive/negative ratio is maintained as 1:3. To do that, we firstly select positive examples according to the results of all positive RoIs (sorted by loss). Then we select false positive RoIs as hard negative examples, and we supplement with negative RoIs that have high loss if it's not enough for the mini-batch.
As mentioned in [19], there is a notice of the selection by sorting loss. In general, co-located RoIs with high overlap are more likely to have similar loss, which induces some areas with high loss to be selected repetitiously. In this regard, the non-maximum suppression (NMS) [19] is also used in our method and the threshold of IoU is set to 0.7.
Update-stage. Since we divide an SGD iteration into two stages, to meet the requirement of bootstrapping and select effective examples, the network is set to not propagate backward or update in the fixed-stage (learn rate is set to 0). In the update-stage, we use the mini-batch constructed in fixed-stage to train the network and update the weights by SGD and multi-task loss. The two stages implement with same layers in the network. To ensure the same weights at the beginning of each iteration, we share the weights of update-stage with the fixed-stage after each iteration so that the network can harvest hard examples dynamically and be trained iteratively.

3.3. Multi-task loss optimization

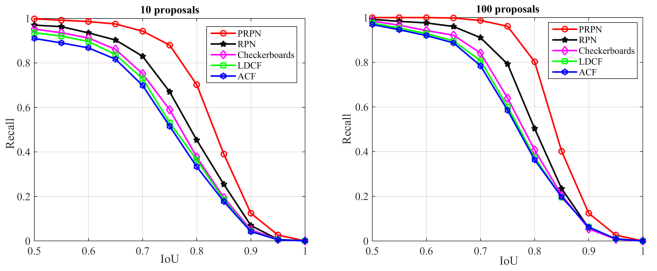
The multi-task loss was firstly proposed in Fast R-CNN [20] and still used in Faster R-CNN for training the network. To exploit the dynamic iterative bootstrapping method with more insights, we investigate the applicability of the multi-task loss to it. The results show that the loss of classification and bbox regression in pedestrian detection are not compatible as they are in general object detection. Since pedestrian detection is a single category task, the classification is relatively simple, which reduces the influence of classification to far less than bbox regression. To resolve the issue, we present a smooth square L_1 loss by combining the square method with smooth L_1 loss used in Faster R-CNN, and utilize it to replace the previous bbox regression loss in multi-task loss. The smooth square L_1 loss is defined as:

$$L_{square}(x) = \begin{cases} 0.5(A \cdot x^2 \cdot \delta)^2 & if|x| < 1/\delta \\ 0.5(|x| - A/\delta)^2 & otherwise \end{cases} \quad (1)$$

where A and δ are hyper parameters for controlling the convergence speed and the weight in the overall loss. We set A to 0.5 and δ to 3 after experimental adjustments.
Then we build the multi-task loss with the log-loss for classification and smooth square L_1 loss for bbox regression as in Faster R-CNN. Compared with the original one used in Faster R-CNN, our improved multi-task loss has faster convergence speed for large loss and is more robust for small loss, as shown in Fig.3(a). Fig.3(b) also indicates that the positive effect of our optimization in the classifier training is obvious compared with Faster R-CNN.

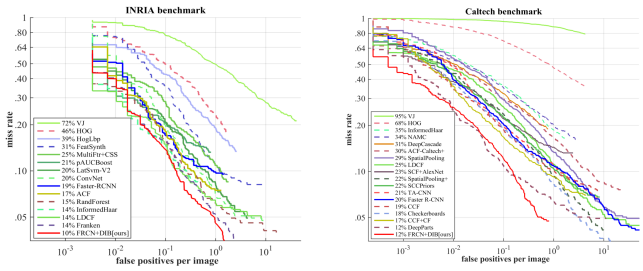
3.4. Implementation details

There are different implementations of bootstrapping in deep networks [19, 21, 22] and each of them has trade-offs. As the dynamic iterative bootstrapping has two stages in each iteration, we implement the network with two branches as shown in Fig.1. The training follows *image-centric* principle [14] that only uses an image at each iteration. For an SGD iteration, the feature map is firstly computed by VGG16-net, then the proposals are generated by PRPN. Although it is feasible to use all proposals to train the classifier, it is unnecessary. To increase efficiency, we also apply NMS to proposals to reduce its number. Then, the dynamic iterative bootstrapping schema is implemented by two branches with the same layers. The selection of hard examples and the construction of mini-batch follow the rules in sec.3.2. The two stages conduct in each iteration at the entire training process.



(a) 10 proposals (b) 100 proposals

Fig. 4. The comparison of PRPN, RPN and three leading algorithms in terms of proposal quality (recall *vs.* IoU).



(a) INRIA benchmark (b) Caltech benchmark

Fig. 5. The comparison of our method (FRCN+DIB) with others state-of-the-art methods.

4. EXPERIMENTS

We comprehensively evaluate our approach on two widely used benchmark datasets: Caltech [17] and INRIA [6]. The Caltech dataset contains 42782 images, where all data and annotations are saved as videos with resolution of 640×480 pixels. The evaluation is performed on *reasonable* standard, which filters too small targets for training and testing. The INRIA dataset contains 902 positive images with people targets, where 614 images are for training and others are for testing. For evaluation, we use the original images that have different sizes but with high-definition.

4.1. Evaluation of PRPN

In this subsection, we evaluate the performance of PRPN in terms of proposals quality on INRIA dataset. For hyper-parameters in PRPN, we set the single ratio r to 2.44, similar value was used for ground-truth annotations in Caltech dataset. The multi-scales schema is set to $16 \times 1.4^{[1,2,3,4,5,6,8,10]}$, which includes a wider range of scales than original RPN. The evaluation metric is recall rates under different IoU thresholds. We evaluate the PRPN on average 10, 100 proposals per image and compare with original RPN and three leading methods based on traditional features: Checkerboards [23], LDCF [13] and SCF [8].

As shown in Fig.4, benefiting from the power of learning feature, the performance of RPN can be comparable to leading traditional methods, but its advantage is not as obvious as in general object detection. Compared with other methods, our PRPN shows state-of-the-art performance and a noticeable improvement over RPN. With 100 proposals per image, our PRPN achieves 96.8% recall at the common IoU of 0.7. The results demonstrate that the relevant adjustments for pedestrian detection is very effective.

4.2. Comparison with state-of-the-art methods

We compare our detector (*FRCN+DIB*) with common state-of-the-art methods on different dataset. The comparison includes leading traditional detectors and deep learning methods. Besides, we deliberately add Faster R-CNN for a clearer comparison. The metric of evaluation is log-average miss rate (MR) on false positive per image and the average MR of the area under the curve.

As the results shown in Fig.5, our detector achieves the state-of-the-art performance and outperforms popular detectors used in pedestrian detection. Our proposed approach is much better than leading traditional detectors, and compared with recent hybrids that combine hand-crafted features with CNN, our approach is still in a superior position.

On the other hand, we note that our detector and CCF [24] are the only methods that use no hand-crafted features, and our method is better than CCF (improves about 5% on Caltech dataset). It is worth mentioning that despite the Faster R-CNN obtains comparative performance, our approach still has a remarkable improvement (about 7%) over it, which demonstrates the efficiency of dynamic iterative bootstrapping in pedestrian detection.

5. CONCLUSIONS

This paper has presented a pedestrian detector based on Faster R-CNN. We introduce a pedestrian region proposal network to generate higher quality candidate regions for detection. To settle the data imbalance issue in the classifier training and resolve the confusions of complex backgrounds, the dynamic iterative bootstrapping strategy is proposed, with the purpose to efficiently utilize hard negative examples. Experimental results on different standard datasets demonstrate the effectiveness of the proposed approach.

6. ACKNOWLEDGMENTS

This work is supported by the Shenzhen Municipal Development and Reform Commission (Disciplinary Development Program for Data Science and Intelligent Computing), and the Shenzhen Engineering Laboratory of Broadband Wireless Network Security.

7. REFERENCES

[1] Piotr Dollár, Serge J Belongie, and Pietro Perona, "The fastest pedestrian detector in the west," in *British Machine Vision Conference*, 2010, vol. 2, p. 7.

[2] Oncel Tuzel, Fatih Porikli, and Peter Meer, "Pedestrian detection via classification on riemannian manifolds," *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 10, pp. 1713–1727, 2008.

[3] Markus Mathias, Rodrigo Benenson, Radu Timofte, and Luc Van Gool, "Handling occlusions with franken-classifiers," in *International Conference on Computer Vision*, 2013, pp. 1505–1512.

[4] 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*, vol. 32, no. 9, pp. 1627–1645, 2010.

[5] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[6] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition*. IEEE, 2005, vol. 1, pp. 886–893.

[7] Pierre Sermanet, Koray Kavukcuoglu, Soumith Chintala, and Yann LeCun, "Pedestrian detection with unsupervised multi-stage feature learning," in *Computer Vision and Pattern Recognition*, 2013, pp. 3626–3633.

[8] Rodrigo Benenson, Mohamed Omran, Jan Hosang, and Bernt Schiele, "Ten years of pedestrian detection, what have we learned?," in *European Conference on Computer Vision*. Springer, 2014, pp. 613–627.

[9] Yonglong Tian, Ping Luo, Xiaogang Wang, and Xiaoou Tang, "Pedestrian detection aided by deep learning semantic tasks," in *Computer Vision and Pattern Recognition*, 2015, pp. 5079–5087.

[10] Bin Yang, Junjie Yan, Zhen Lei, and Stan Z Li, "Aggregate channel features for multi-view face detection," in *International Joint Conference on Biometrics*. IEEE, 2014, pp. 1–8.

[11] Yonglong Tian, Ping Luo, Xiaogang Wang, and Xiaoou Tang, "Deep learning strong parts for pedestrian detection," in *International Conference on Computer Vision*, 2015, pp. 1904–1912.

[12] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," in *Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

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

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Neural Information Processing Systems*, 2015, pp. 91–99.

[15] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[16] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] Piotr Dollar, 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*, vol. 34, no. 4, pp. 743–761, 2012.

[18] Kah Kay Sung, "Learning and example selection for object and pattern detection," 1995.

[19] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick, "Training region-based object detectors with online hard example mining," in *Computer Vision and Pattern Recognition*, 2016, pp. 761–769.

[20] Ross Girshick, "Fast r-cnn," in *International Conference on Computer Vision*, 2015, pp. 1440–1448.

[21] Ilya Loshchilov and Frank Hutter, "Online batch selection for faster training of neural networks," *arXiv preprint arXiv:1511.06343*, 2015.

[22] Xiaolong Wang and Abhinav Gupta, "Unsupervised learning of visual representations using videos," in *International Conference on Computer Vision*, 2015, pp. 2794–2802.

[23] Shanshan Zhang, Rodrigo Benenson, and Bernt Schiele, "Filtered channel features for pedestrian detection," in *Computer Vision and Pattern Recognition*. IEEE, 2015, pp. 1751–1760.

[24] Bin Yang, Junjie Yan, Zhen Lei, and Stan Z Li, "Convolutional channel features," in *International Conference on Computer Vision*, 2015, pp. 82–90.