Aspect Ratio Sensitive Network (ARS-Net)

Project Report of CS272 Computer Vision II

Jianxiong Cai
ShanghaiTech University
caijx@shanghaitech.edu.cn

Anqi Pang
ShanghaiTech University
pangaq@shanghaitech.edu.cn

Peijia Xu
ShanghaiTech University
xupj@shanghaitech.edu.cn

Lei Jin
ShanghaiTech University
jinlei@shanghaitech.edu.cn

Ruijian Li
ShanghaiTech University
lirj@shanghaitech.edu.cn

Abstract

Many objects in the real world have a prior knowledge, which could be helpful for object detection. In this project, we propose an approach to include aspect ratio as a prior knowledge. Our approach duplicates the RPN network in faster RCNN [16] so that different RPN can focus on generating proposal of different shapes.

Our approach dramatically improves the performance (AP) of certain classes. However, our approach brings two major drawbacks: 1) the speed decreases as adding more RPNs. 2) more false negative appear as different RPNs generate an equal number of proposals for now.

1. Introduction

General-purpose object detection on RGB images plays an important role in various applications, like autonomous driving and security. Two-stage approach like Faster RCNN [16] shows a better performance in terms of mAP, compared with one-stage network such as YOLO [15].

In Faster RCNN, RPN Network is used for proposing ROI regions. Lots of work has been made to refine region proposal and combine prior knowledge, which would be introduced in section 2. In this project, we include the prior knowledge of aspect ratio.

As objects usually do not deform so much, they generally have a relatively fixed shape in some extent, which infers that the target bounding box will have a certain relationship with aspect ratio, especially for non-living objects. An intuitive example is that bottle will be more likely to have large aspect ratios (width to height), i.e. tall and thin. According to the statistics, both of the two commonly used datasets, VOC [3] and MS COCO [13], are showing the regulation. Following histograms describe this phenomenon for some classes, where objects tend to have some different aspect ratios.

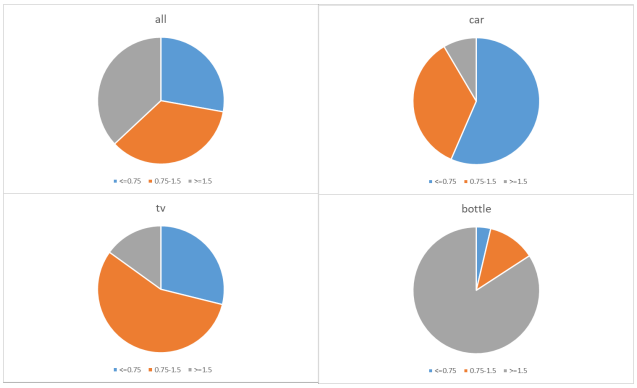

Figure 1. Pie charts of MS COCO[13], where a car (top right) tends to be long, tv (down left) square and bottle (down right) tall. Same can be observed on VOC07[3] dataset.

The original RPN network in Faster RCNN [16] has 3 anchor shapes: 1:2, 1:1 and 2:1. Thus we count the distribution of object shape by dividing their aspect ratios into those three ratios, as is shown in fig.1. One may conclude that in both datasets, car (top right) tends to be long, tv (down left) square and bottle (down right) tall. This justifies our assumption about the prior knowledge of aspect ratio distribution.

In this project, we propose a new approach (as shown in fig.2) to include aspect ratio as the prior knowledge for object detection. Although aspect ratio is a relatively weak feature for objects, we observe that it holds for certain object classes in most major datasets. Our approach dramatically improves the performance (AP) for those certain classes.

2. Related Work

R-CNN Girshick et al. [6] addressed the R-CNN detector which promotes the accuracy of neural networks for ob-

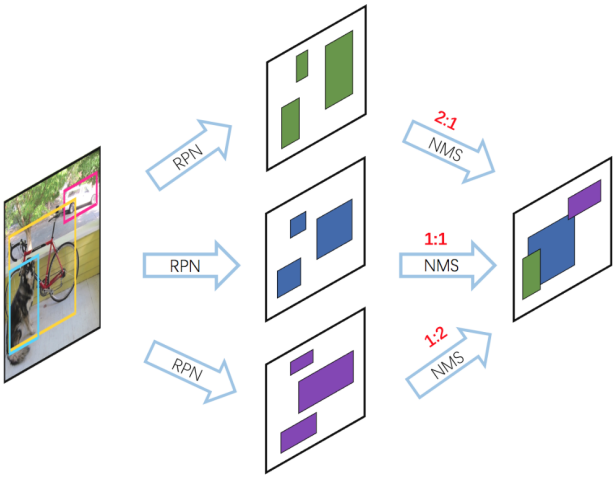

Figure 2. ARS-Net RPN architecture

ject detection. This R-CNN detector combines a region proposal part (Selective Search [18]), a CNN feature extractor and a classifier. R-CNN detector surmounts conventional detectors in the aspect of accuracy but its speed is limited by generating massive region proposals and extracting features using CNN on each of the region proposals. He et al. [8] later introduced a way to meliorate by employ pyramid pooling layer. SPP-net speeds up the detector by computing CNN only once per image. Imitating SPP-net, Fast-RCNN [5] shown that using multi-task learning and back-propagation through ROI pooling layer will speed up the R-CNN detector by an order of magnitude. The bottleneck of Fast-RCNN is conventional region proposal generator, so Faster-RCNN [16] proposed Region Proposal Network (RPN) which sharing features with classifier network.

Region Proposal Network Most traditional region proposal methods are based on low-level features. Faster-RCNN introduced Region Proposal Network (RPN) to generate region proposals based on high-level semantic features. RPN surpassed conventional region proposal methods, either unsupervised method (e.g. Selective Search [18] and EdgeBoxes [24]) or supervised method (e.g. BING [2]), on both speed and accuracy.

Cascade Refinement Region Proposal Network can be refined by applying a multi-stage cascading pipeline structure. Applying a two-class Fast RCNN to refine proposals has shown significant improvement. [20] [22] [9]. Ghodrati et al. [4] introduced a Cascading Deep Convolutional Layers to get feature maps. Cascaded CNN [7] proposes a RefineNet added after RPN which can effectively reduce the number of proposals and improve their confidence. Xiang and Zhu [19] apply similar approach, using a R-net to refine P-net. Li et al. [10] introduce a spatial correlation related method, using a new EM-like group recursive learning approach to iteratively refine object proposals and provide an optimal spatial configuration of object detections.

Multi-scale Refinment Methods like SDP,SSH or MS-CNN [21] [14] [1]make independent predictions at different layers, which will ensure smaller objects are trained on higher resolution layers. Other methods like FPN [11], RetinaNet [12] proposed a pyramidal structure to combine shallow layer with deep layer to get both high-level feature and low-level feature.

ROI and NMS Zhu et al. [23] proposed a two-branch network of ROI after RPN to obtain both local part information and global information.

3. Aspect Ratio Sensitive Object Detection

3.1. Network Architure

Multi-RPN To include aspect ratio as a prior knowledge for object detection, we modified the network architecture by duplicating the region proposal network into 3 copies, each handling objects of different shapes. During training (see figure 3), green RPN (right) is only provided with ground truth bounding boxes of aspect ratio greater than 2:1, grey one (middle) with boxes around 1:1 only and yellow one (left) with boxes smaller than 1:2 only. Results of three region proposal networks are simply stacked together in testing time. In this way, our network is able to learn separate filters for objects of different shapes. Non-maximum suppression is applied in the end to reject high over-lapping objects.

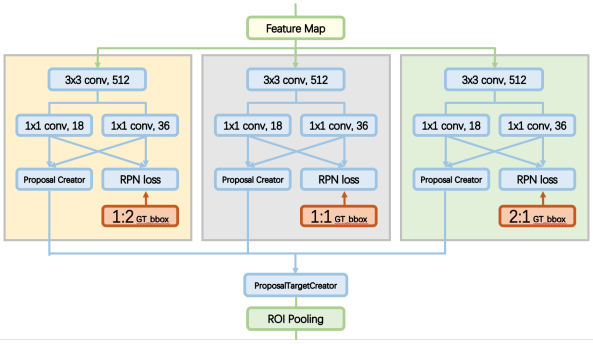

Figure 3. Multi-RPN approach

Sharing Features The multi-RPN network works, but comes at great cost. One of the obvious shortcomings of this approach would be the high overall time complexity. Compared with the original faster RCNN, which runs at about 9fps on one Nvidia TitanX, our multi-RPN approach can

only run at half speed. Sharing features, plus adding additional one-by-one convolutional layers are applied to cope with such problem. More specifically (see Figure 4), three RPNs share the same 3*3*512 layer, but each would have another 1*1*256 layer of its own to further reduce depth. Other parts of the network remains unchanged. From table 1, adding 1*1 conv into our network improves speed while maintaining a similar mAP. However, it's still slow when compared with the origin RCNN, after all, region proposal network still remains to be the biggest overhead of faster RCNN.

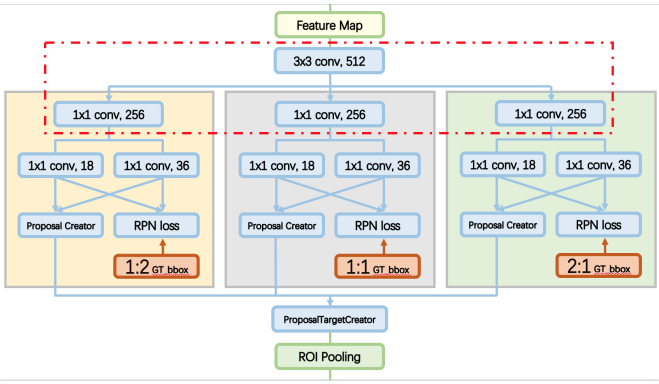

Figure 4. Sharing Features

Approach	Frame per second
faster RCNN (300 proposals)	9
ARS-Net (300 proposals)	4.73
ARS-Net (900 proposals)	3.56
ARS-Net+1*1 conv (300 proposals)	5.04
ARS-Net+1*1 conv (900 proposals)	3.84

Table 1. Time complexity under different settings, all reported on one Nvidia TitanX. Adding 1*1 conv into our network improves speed while maintaining a similar mAP.

4. Experiment

We evaluate our method on the PASCAL VOC 2007 detection benchmark. This dataset consists of 5011 train images and 4952 test images over 20 object categories. Evaluation is performed on all test images. We calculated AP for every single class, and mAP at last. This is also a general metric in object detection. PR curve is also included in figure 5.

4.1. Backbone Network

We set VGG16[17] as our backbone network, which has 13 convolutional layers followed by 3 fully-connected layers. This is the same setting with the origin faster RCNN[5].

4.2. Training Details

In general, we adopted the training approaches from the origin faster RCNN. Convolutional layers are drawn from a zero-mean Gaussian distribution with standard deviation with 0.01, except in the first 13 feature extraction layers[16]. During training, we set the parameters of the first and the second stage of the backbone network as fixed. One single image is chosen randomly at each step to provide positive and negative samples up to 1:3 ratio for both RPNs and final classifiers. We trained the network with stochastic gradient descent (SGD) end-to-end, the learning rate is set to 0.001 for the first 15 epochs. Then we fine-tuned the model with the best accuracy with another learning rate of 0.0001 for the next 15 epochs.

4.3. Experiments on PASCAL VOC

Table 2 shows the results on VGG-16[17] model. Categories are highlighted if we are able to reach a better performance. Using ARS-Net, the result is 73.2% for sharing features between proposal and detection, compared to 71.8% that evaluated on single RPN. As shown above, our method has also dramatically improved AP in certain categories such as bottle (by 3.9%) and cow (by 5.9%), which has a strong prior in shape. This is because we specify each RPN processing selected ground truth during training, which leads to the aspect ratio sensitivity. Thus the proposals generated by ARS-Net for those categories are more accurate. There's a slight decrease in several classes, which is a side effect. One possible reason is that the distribution of the test set is not consistent with the training set.

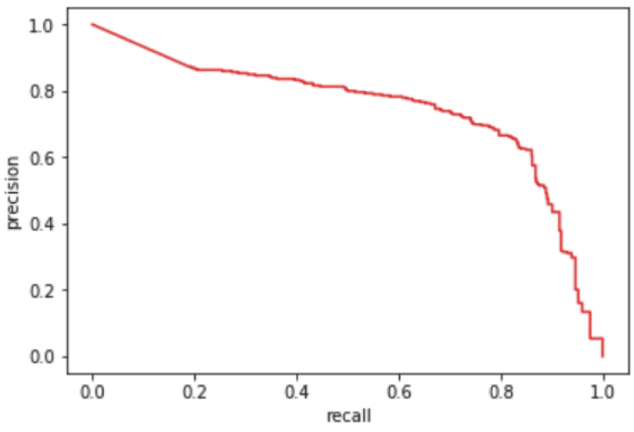

Figure 5. PR curve

4.4. Ablation Experiments

We further tested ARS-Net with a diverse number of proposals, which is as shown in table 3. When the number of proposals drops from 900 (Top 300 proposals are selected

method	# of proposals	data	mAP	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Faster RCNN	300	VOC07	71.8	73.5	81.5	68.5	53.7	52.3	80.7	85.3	84.3	52.5	76.8	71.5	81.3	84.9	75.1	79.6	44.7	72.5	65.9	79.8	72.3
ARS-Net	900	VOC07	**73.2**	**74.5**	81.5	**71.1**	**56.7**	**56.2**	**83.6**	**85.8**	**87.4**	51.4	**82.7**	68.9	**83.8**	**86.5**	**77.1**	**81.0**	43.9	**73.0**	**66.4**	79.7	**72.6**

Table 2. Average precision of different classes reported on origin faster RCNN and out ARS-Net, highlighted ones are categories with a higher mAP

from each scale) to 300, mAP reduces by 1.7%. And a further decrease in the number of proposals leads to dropping on mAP. The main reason we thought might be that more proposals mean we may have multiple proposals for a single object, where the best proposal will be selected during none maximum suppression by score. In fact, when the number of proposals is 1200 in total (400 each), mAP gets a slight increase (From 73.2% to 73.3%). However, this change does more harm than good. It will lead to a decrease in fps. So we take 900 proposals in total as our final result. This maintains a relatively high mAP while also taking speed into account. After using the trick of sharing features, we make ARS-Net as fast as original Faster R-CNN.

method	# of proposals	mAP
ARS-Net	300	71.5
ARS-Net	900	73.2
ARS-Net	1200	73.3

Table 3. Increasing number of proposals in ARS-Net

5. Things we tried but didn't work

As mentioned above, one of the major drawbacks with ARS-NEt is that it produces more false positive because the three sub-networks propose an equal number of proposals. For some images, all objects may concentrate on 1*1 shape, which means most proposals from other two networks are false positive.

As the result, we tried different approaches to suppress those proposals.

Additionally, we want to take full advantage of aspect ratio in other parts but fail, unfortunately.

5.1. Add NMS after RPNs.

Three RPNs means more region proposal and lead more false positives in our detection results as for original faster R-CNN will decrease some RoIs in the same locations with different shapes by NMS in RPN(it might be right for some cases but not always).

We simply apply NMS without score and lead mAP decrease by 0.5 roughly, decrease by 1 approximately when choosing 300 RoIs after NMS. To avoid the influence caused by order, we shuffle the RoIs after stacking them all together. Removing shuffle will make mAP down much more. The problem for NMS only depending on IoU may cause both false positives and negatives, as NMS can't recognize which is the right one to be left.

When applying NMS with score, we face a trouble that the three RPNs are scoring the RoIs independently so its hard to make a comparison and there is also a question that if the scores should be normalized so we didnt implement the NMS with score in this step.

5.2. Rescoring Based on Prior Distribution

The idea to reduce score of bounding boxes with rare aspect ratios is naive. We simply multiply predicted score of the probability(calculate from the statistics of dataset) of the aspect ratio occurred in its corresponding class. It leads more than 10 decreases although the thought seems to make sense at first glance.

We try to multiply the predicted score with the prior possibility and a constant k but it doesn't work as well.

It failed mainly because it is almost impossible to design a rescore mapping by hand and it is not robust for extreme cases. What's more, the rightness of prior possibility is questioning.

5.3. Cascade Head Part

All the tries can be summarized as duplicate removal indeed. The initial idea is to reshape the bounding boxes generated in the first head by prior ratio distribution to convey the class information to next head but fail to design a suitable reshape mapping based on aspect ratio. Then we try cascade purely but maybe some faults in programming or mistakes in training lead that cascade method failed during training and it actually doesn't relate much to aspect ratio.

We try to decrease IoU of NMS and increase the lower bound of score when implementing the second head but failed. Additionally, we also try to weight the loss of different head to make the second head a more accurate classifier and regressor but fail while training. (The loss function diverges at the first epoch and then we reduce the learning rate but the mAP of the first epoch is below 10 so we quit).

5.4. Hard-negative Mining

We tried hard-negative mining on VOC2007 (both on original network architecture and multi-RPN network). However, we didn't get obvious improvement with either network.

6. Discussion

Although aspect-ratio has been proved to be effective on current major datasets like VOC2007[3] and MSCOCO[13], it is relatively a weak feature for real-world

objects. For example, objects like tables may show different shapes in different viewpoints. Some more strong features might be more effective for object detection.

7. Conclusion

By including aspect ratio as a prior knowledge, our network dramatically improves performance on detecting aspect-ratio sensitive objects. However, on other objects, the performance reduce because of more false positive.

8. Contribution

Following are the contribution from each team member.

1. Jianxiong Cai: Training, hard-negative mining and Coordination

2. Jinglei: Multi-RPN, Feature Sharing, Training

3. Ruijian Li: Final stage suppression, testing & organizing.

4. Peijia Xu: Dataset distribution analysis, cascade head, NMS after RPNs and rescoring(failed :( )

5. Anqi Pang: Idea, Survey Related works, some Testing and NMS and final checking.

References

[1] Zhaowei Cai, Quanfu Fan, Rogério Schmidt Feris, and Nuno Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. *CoRR*, abs/1607.07155, 2016. URL http://arxiv.org/abs/1607.07155.

[2] M. M. Cheng, Z. Zhang, W. Y. Lin, and P. Torr. Bing: Binarized normed gradients for objectness estimation at 300fps. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3286–3293, June 2014. doi: 10.1109/CVPR.2014.414.

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[4] A. Ghodrati, A. Diba, M. Pedersoli, T. Tuytelaars, and L. V. Gool. Deepproposal: Hunting objects by cascading deep convolutional layers. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2578–2586, Dec 2015. doi: 10.1109/ICCV.2015.296.

[5] R. Girshick. Fast r-cnn. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, Dec 2015. doi: 10.1109/ICCV.2015.169.

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '14, pages 580–587, Washington, DC, USA, 2014. IEEE Computer Society. ISBN 978-1-4799-5118-5. doi: 10.1109/CVPR.2014.81. URL https://doi.org/10.1109/CVPR.2014.81.

[7] Y. Guo, X. Guo, Z. Jiang, and Y. Zhou. Cascaded convolutional neural networks for object detection. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4, Dec 2017. doi: 10.1109/VCIP.2017.8305026.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, Sept 2015. ISSN 0162-8828. doi: 10.1109/TPAMI.2015.2389824.

[9] W. Kuo, B. Hariharan, and J. Malik. Deepbox: Learning objectness with convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2479–2487, Dec 2015. doi: 10.1109/ICCV.2015.285.

[10] J. Li, X. Liang, J. Li, Y. Wei, T. Xu, J. Feng, and S. Yan. Multi-stage object detection with group recursive learning. *IEEE Transactions on Multimedia*, pages 1–1, 2017. ISSN 1520-9210. doi: 10.1109/TMM.2017.2772796.

[11] T. Y. Lin, P. Dollr, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, July 2017. doi: 10.1109/CVPR.2017.106.

[12] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollr. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, Oct 2017. doi: 10.1109/ICCV.2017.324.

[13] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[14] M. Najibi, P. Samangouei, R. Chellappa, and L. S. Davis. Ssh: Single stage headless face detector. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4885–4894, Oct 2017. doi: 10.1109/ICCV.2017.522.

[15] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. *arXiv preprint arXiv:1612.08242*, 2016.

[16] 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*, 39(6):1137–1149, June 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2577031.

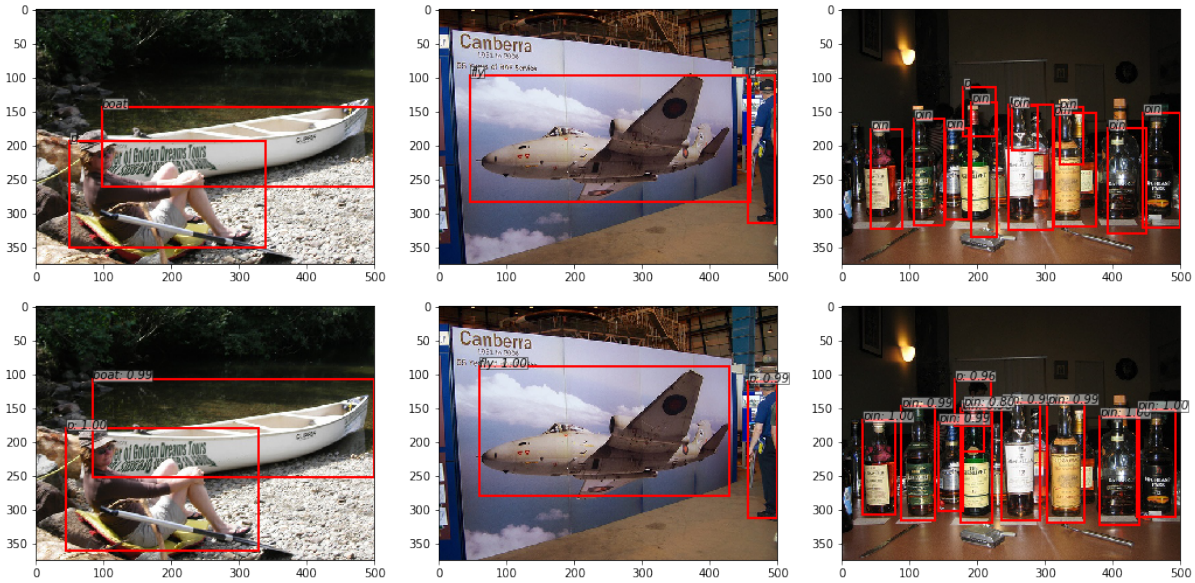

Figure 6. A comparsion of the results between origin faster RCNN (above) and our approach (below). We are able to achieve a tighter and better bounding box

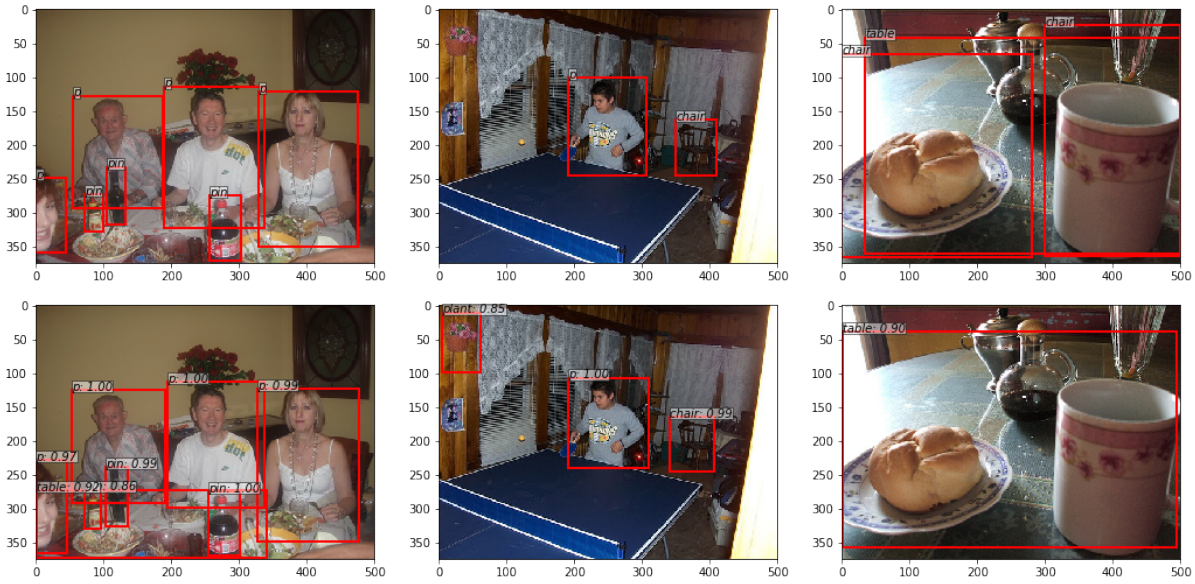

Figure 7. A comparsion of the results between origin faster RCNN (above) and our approach (below). We are able to achieve a tighter and better bounding box

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

[18] K. E. A. van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders. Segmentation as selective search for object recognition. In *2011 International Conference on Computer Vision*, pages 1879–1886, Nov 2011. doi: 10.1109/ICCV.2011.6126456.

[19] J. Xiang and G. Zhu. Joint face detection and facial expression recognition with mtcnn. In *2017 4th International Conference on Information Science and Control Engineering (ICISCE)*, pages 424–427, July 2017. doi: 10.1109/ICISCE.2017.95.

[20] B. Yang, J. Yan, Z. Lei, and S. Z. Li. Craft objects from images. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6043–6051, June 2016. doi: 10.1109/CVPR.2016.650.

[21] F. Yang, W. Choi, and Y. Lin. Exploit all the layers: Fast and

accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2129–2137, June 2016. doi: 10.1109/CVPR.2016.234.

[22] X. Zeng, W. Ouyang, J. Yan, H. Li, T. Xiao, K. Wang, Y. Liu, Y. Zhou, B. Yang, Z. Wang, H. Zhou, and X. Wang. Crafting gbd-net for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2018. ISSN 0162-8828. doi: 10.1109/TPAMI.2017.2745563.

[23] Y. Zhu, C. Zhao, J. Wang, X. Zhao, Y. Wu, and H. Lu. Couplenet: Coupling global structure with local parts for object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4146–4154, Oct 2017. doi: 10.1109/ICCV.2017.444.

[24] C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 391–405, Cham, 2014. Springer International Publishing.