PSRR-MaxpoolNMS: Pyramid Shifted MaxpoolNMS with Relationship Recovery – Supplementary Material

Tianyi Zhang[1] Jie Lin[1*] Peng Hu[2] Bin Zhao[3] Mohamed M. Sabry Aly[4]
[1] I2R, A*star, Singapore [2] Sichuan University, China [3] IME, A*star, Singapore [4] NTU, Singapore
{zhang_tianyi, lin-j}@i2r.a-star.edu.sg, penghu.ml@gmail.com, zhaobin@ime.a-star.edu.sg, msabry@ntu.edu.sg

In this supplementary material, we first introduce the detailed definition of Box Overlap in Section A. Section B provides pilot experiments to verify the limitations of MaxpoolNMS [1], which motivate us to develop our PSRR-MaxpoolNMS. In addition, we perform ablation studies on the effect of scale channels for our Channel Relationship Recovery in Section C.

A. Definition of Box Overlap

The computation of Box Overlap is defined as follows: for each box category of each image, given the input box candidates (e.g. 300 boxes at the second stage of Faster-RCNN), we feed them into GreedyNMS and our PSRR-MaxpoolNMS to obtain the indices of the final selected boxes. Then we calculate intersection set and union set of the two set of indices from GreedyNMS and our PSRR-MaxpoolNMS. For each box category, the overlap is computed as the intersection over the union. We finally report the averaged overlap of all the categories as our reported Box Overlap. Our Box Overlap is similar to the mean intersection over union (mIOU) criteria in semantic segmentation, but only differs in the definitions of intersection and union.

B. Verify the limitations of MaxpoolNMS

Our proposed PSRR-MaxpoolNMS is mainly motivated by the following limitations/assumption of MaxpoolNMS [1] that we have identified: 1) The score map mismatch problem caused by ignoring box regression in the confidence score maps of MaxpoolNMS, hence our Relationship Recovery module is introduced to tackle this issue. 2) The low sparsity of score maps cannot be simply solved by increasing the threshold α. Larger α often leads to higher sparsity (and precision), but decreases the recall rate (true positives removed). 3) The underlying assumption of MaxpoolNMS that overlapped boxes only exist in the channels with adjacent scales (or ratios) on the score maps is not always true. The overlapped boxes can be distributed at arbitrary scales and ratios. Thus, our Pyramid Shifted MaxpoolNMS is proposed to increase the score map sparsity and detection precision, without hurting recall and without any assumption on the distribution of overlapped boxes. In this section, we provide experimental analysis to verify the limitations/assumption of MaxpoolNMS.

B.1. Experimental Setup

All the following experiments are performed at the second stage of Faster-RCNN pipeline with ResNet-50 backbone on PASCAL VOC or KITTI dataset. There are 300 boxes with refined confidence scores and box locations generated by the second stage per image per category, before the NMS post processing.

Choice of Scale Channels. In particular, following the default training parameters of the public PyTorch implementation of Faster-RCNN [1], we simply set the anchor scales used for training as $\left[128^2, 256^2, 512^2\right]$ on both PASCAL VOC and KITTI datasets. In our Channel Recovery step during inference, we set the channels in scale as $\left[64^2, 128^2, 256^2, 512^2\right]$. We add the small scale channel (*i.e.* 64^2) in consideration of box regression that boxes are regressed to smaller scales (e.g. 128^2 to 64^2). Thus, boxes that have been scaled down after regression are projected to small scale channel, which in turn reduces the risk of suppressing these small boxes wrongly due to a large max pooling kernel size, as evidenced in the following experiments.

B.2. Evaluation of Score Map Mismatch

We investigate the score map mismatch problem caused by box regression effect. All statistics are aggregated from the 300 boxes before NMS over all categories of all evaluation images.

We measure the **Spatial Mismatch** as the average normalized shift of box centers before and after box regression.

[1] https://github.com/jwyang/faster-rcnn.pytorch

	64×64	128×128	256×256	512×512
128×128	18.95%	17.92%	3.31%	0.03%
256×256	0.01%	4.45%	26.83%	3.22%
512×512	0.00%	0.02%	10.40%	14.86%

Figure A. Scale Mismatch measured by transition probability matrix for scale channels on PASCAL VOC dataset. Vertical axis indicates original anchor scales, horizontal axis indicates the recovered scale channels of the regressed boxes.

	0.5	1	2
0.5	17.96%	9.17%	0.24%
1	6.40%	16.01%	3.34%
2	5.79%	22.36%	18.73%

Figure B. Ratio Mismatch measured by transition probability matrix for ratio channels on PASCAL VOC dataset. Vertical axis indicates anchor ratios, horizontal axis indicates the recovered ratio channels of the regressed boxes.

In more detail, the normalized shift is calculated by the center pixel absolute difference between the regressed box and the corresponding anchor box, divided by the anchor size (anchor width or anchor height). Results are reported in Table A. We observe that box regression introduces dramatic spatial shifts (shifts of 15% to 20% of the anchor dimensions), leading to the spatial mismatch problem.

Table A. Statistics of the Spatial Mismatch caused by box regression on PASCAL VOC dataset. The spatial mismatch is measured as average normalized shift of box centers before and after box regression.

-	horizontal	vertical
avg-normalized shift (%)	16.5	17.0

As for the **Channel (scale and/or ratio) Mismatch**, we report the so-called transition probability matrix for the scale channels (see Figure A) and ratio channels (see Figure B), respectively. For each box we retrieve the scale (ratio) channel of its initial anchor and the scale (ratio) channel of the regressed box after Channel Recovery, which forms the transition matrix. Each element in the transition matrix represents the probability of transition from one scale (ratio) to another. Figure A shows that anchor scales are likely to be regressed to adjacent scale channels. Approximately 40% boxes are regressed into different scale channels. It is worth noting that almost half of the 128^2 anchor boxes are regressed to smaller scale 64^2, implying the necessity of adding 64^2 scale channel in our Channel Recovery step. Figure B shows that anchor box ratios are possibly to be regressed to arbitrary ratio channels. Approximately 50% boxes are regressed into different ratio channels.

B.3. Evaluation of Sparsity with α

In this section, we show that simply increasing α can produce higher sparsity on score maps, but it cannot perform on par with our Pyramid Shifted MaxpoolNMS in terms of detection accuracy. We experiment on varied overlap threshold α, with different channel combinations for single-scan max pooling after our relationship recovery. Results are reported in Figure C. We observe that single-scan max pooling performs better as α increases, but the performance drops if α is too large, regardless of the channel combinations used. This is probably because a larger α is more likely to suppress true positive boxes wrongly. Therefore, simply increasing α performs significantly worse than our Pyramid Shifted MaxpoolNMS, e.g. the highest mAP achieved by Cross-Ratio MaxpoolNMS (ratio) is 70.6% (with α=1.5), versus 77.6% achieved by our Pyramid Shifted MaxpoolNMS.

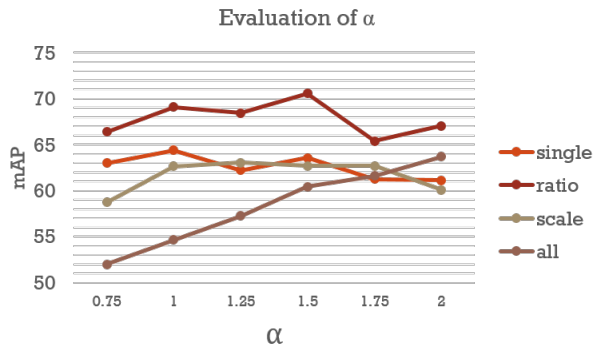

Figure C. Detection accuracy (mAP) as a function of the overlap threshold α, with different channel combinations for single-scan max pooling after relationship recovery on PASCAL VOC dataset. The highest mAP achieved by Cross-Ratio MaxpoolNMS (ar) is 70.6% (with α=1.5). For reference, the mAP of our Pyramid Shifted MaxpoolNMS is 77.6%, which outperforms all the single-scan max pooling with any α.

B.4. Scale/Ratio Distribution of Overlapped Boxes

In this section, we validate that boxes with large overlap (IoU>0.3) can be distributed at arbitrary scales and ratios on the score maps, rather than the assumption that they only exist in channels with adjacent scales (or ratios). For each box category of each image, we calculate the IoU score between each pair of 300 candidate boxes and record the scale/ratio channel of each largely overlapped pair with IoU>0.3. Figure D and Figure E report the distribution of these overlapped pairs over scale and ratio channels respectively. We observe that overlapped pairs can be distributed across adjacent scale channels and ar-

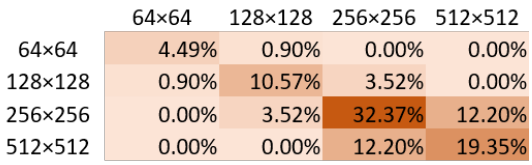

Figure D. Distribution of overlapped box pairs with IoU>0.3 over scale channels on the confidence score maps generated from PASCAL VOC dataset.

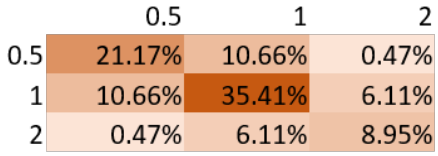

Figure E. Distribution of overlapped box pairs with IoU>0.3 over ratio channels on the confidence score maps generated from PASCAL VOC dataset.

bitrary ratio channels. We also report the distribution of largely overlapped pairs over all combinations of scale and ratio channels in Figure F. One can see that box pairs with large overlap occur almost cross arbitrary channels (except the ones with large differences in scale channels). Thus we propose to operate max pooling with different channel combinations (*i.e.* , Single-Channel MaxpoolNMS, Cross-Ratio MaxpoolNMS, Cross-Scale MaxpoolNMS, Cross-all-Channel MaxpoolNMS) on the score maps sequentially, in order to increase score map sparsity by suppressing overlapped box pairs progressively. We also notice that boxes with large differences in scale (*eg.* , 64^2 and 512^2) are unlikely to be highly overlapped (*i.e.* , IoU>0.3). Thus, max pooling over all channels in our Cross-all-Channel MaxpoolNMS could possibly suppress the true positives wrongly. However, the probability is very low since the kernel size is small for the max pooling operator of Cross-all-Channel MaxpoolNMS.

C. Ablation Study

C.1. Scale Channels for Channel Recovery

As mentioned in Section B.1, in our Channel Recovery step during inference, the channels in scale are set as $\left[64^2, 128^2, 256^2, 512^2\right]$, with one additional small scale 64^2 added to the anchor scales $\left[128^2, 256^2, 512^2\right]$ used for Faster-RCNN training. In this section, we evaluate the effect of scale channels for Channel Recovery on both PASCAL VOC and KITTI datasets, following the protocols in Section B.1. Detection results are reported in Table B and Table C respectively. We observe that 4-scale performs better than 3-scale especially for the moderate or difficult tasks on KITTI dataset. One reasonable explanation is there

Table B. Effect of scale channels for Channel Recovery on PASCAL VOC dataset. 4-scale and 3-scale denote scale channels are set as $\left[64^2, 128^2, 256^2, 512^2\right]$ and $\left[128^2, 256^2, 512^2\right]$ respectively.

-	3-scale	4-scale
mAP (%)	77.2	77.6

are considerable densely clustered small objects in KITTI dataset, these small objects (e.g. close to the scale 64^2) prevent from being suppressed with the help of the additional scale channel 64^2 which is associated with a small kernel size for the subsequent max pooling on the channel.

C.2. Discussion

It is worth noting that our Channel Recovery does not enforce the anchor scales used for training to be the same as the scale channels used for Channel Recovery during inference. This provides further evidence that our Relationship Recovery module is totally anchor-free, and thus our PSRR-MaxpoolNMS can be applied to both anchor-based and anchor-free convolutional object detectors, which is left for future work.

References

[1] Lile Cai, Bin Zhao, Zhe Wang, Jie Lin, Chuan Sheng Foo, Mohamed Sabry Aly, and Vijay Chandrasekhar. Maxpoolnms: getting rid of nms bottlenecks in two-stage object detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9356–9364, 2019. 1

		64×64	128×128	256×256	512×512	64×64	128×128	256×256	512×512	64×64	128×128	256×256	512×512
		0.5	0.5	0.5	0.5	1	1	1	1	2	2	2	2
64×64	0.5	0.013136	0.002525	0	0	0.002255	0.00046	0	0	5.89E-05	2.13E-05	0	0
128×128	0.5	0.002525	0.027817	0.008176	0	0.001032	0.005306	0.002921	0	5.96E-05	0.000231	0.000138	0
256×256	0.5	0	0.008176	0.064275	0.021549	0	0.00281	0.030551	0.013453	0	0.000214	0.001828	0.000309
512×512	0.5	0	0	0.021549	0.041969	0	0	0.020811	0.026978	0	0	0.001412	0.000409
64×64	1	0.002255	0.001032	0	0	0.01412	0.002153	0	0	0.001917	0.000507	0	0
128×128	1	0.00046	0.005306	0.00281	0	0.002153	0.033766	0.009925	0	0.00064	0.005703	0.002161	0
256×256	1	0	0.002921	0.030551	0.020811	0	0.009925	0.115334	0.044418	0	0.003664	0.02338	0.003213
512×512	1	0	0	0.013453	0.026978	0	0	0.044418	0.077884	0	0	0.01284	0.007088
64×64	2	5.89E-05	5.96E-05	0	0	0.001917	0.00064	0	0	0.009162	0.001587	0	0
128×128	2	2.13E-05	0.000231	0.000214	0	0.000507	0.005703	0.003664	0	0.001587	0.021603	0.005144	0
256×256	2	0	0.000138	0.001828	0.001412	0	0.002161	0.02338	0.01284	0	0.005144	0.03261	0.003968
512×512	2	0	0	0.000309	0.000409	0	0	0.003213	0.007088	0	0	0.003968	0.004687

Figure F. Distribution of overlapped box pairs with IoU$>$0.3 over scale and ratio channels on the confidence score maps generated from PASCAL VOC dataset.

Table C. Effect of scale channels for Channel Recovery on KITTI dataset. 4-scale and 3-scale denote scale channels are set as $\left[64^2, 128^2, 256^2, 512^2\right]$ and $\left[128^2, 256^2, 512^2\right]$ respectively.

Method	mAP(easy to hard)			Car			Pedestrian			Cyclist		
				Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mode	Hard
3-scale (ResNet-50)	92.3	84.5	77.0	98.2	92.0	80.0	85.8	75.4	68.2	92.9	86.1	82.8
4-scale (ResNet-50)	93.4	88.5	82.8	96.4	95.6	87.9	90.1	80.9	74.7	93.6	89.0	85.7
3-scale (ResNet-101)	91.5	84.2	76.6	96.2	91.0	79.0	85.1	74.8	67.7	93.2	86.9	83.2
4-scale (ResNet-101)	93.5	88.1	81.2	95.9	95.5	86.1	89.5	79.5	72.1	95.1	89.1	85.2
3-scale (ResNet-152)	91.9	85.3	77.8	97.5	91.8	78.8	86.4	76.2	70.8	91.7	87.9	83.9
4-scale (ResNet-152)	93.8	89.5	82.7	96.8	96.1	86.9	90.7	82.8	75.6	93.8	89.5	85.6