Adversarially-Aware Robust Object Detector (Supplementary Material)

Ziyi Dong, Pengxu Wei*, and Liang Lin

Sun Yat-Sen University, Guangzhou, China
dongzy6@mail2.sysu.edu.cn, weipx3@mail.sysu.edu.cn, linliang@ieee.org

This supplementary file provides more details on the conflict validation, the evaluation of m in loss change validation, quantitative detection results of SSD, MTD, our RobustDet (without CFR) and RobustDet* for each category, and more clear visualization results of SSD, MTD and our RobustDet (with CFR).

A Evaluation of the Conflict between Clean and Adversarial Images

Algorithm 1: Impact of learning the mini-batch X^a on X^b

Data: mini-batch X^a, X^b, attacker A, training steps m.
Result: change of the loss Δl

$l_{before} \leftarrow \mathcal{L}(\theta, X^b)$; // calculate the loss of X^b before the model learns X^a.
for *i=1* **to** m **do**
  $\delta \leftarrow A_{x \in X^a}(\theta, x)$; // calculate attack perturbations.
  // update model parameters.
  $g_\theta \leftarrow \mu g_\theta - \mathbb{E}_{x \in X^a}\left[\nabla_\theta \mathcal{L}(\theta, x + \delta_x)\right]$;
  $\theta \leftarrow \theta + \gamma g_\theta$;
end
$l_{after} \leftarrow \mathcal{L}(\theta, X^b)$; // calculate the loss of X^b after the model learns X^a.
$\Delta l \leftarrow l_{after} - l_{before}$;

To empirically analyze the conflict between the learning of clean images and adversarial images, we evaluate the changes in loss of the model on mini-batch X^b after learning mini-batch X^a and inspect their impacts of learning two kinds of images, as shown in Algorithm 1. Firstly, the loss l_{before} is calculated by taking images from the mini-batch X^b as model inputs. Then, the model is further trained on images from another mini-batch X^a for m steps. With the derived model, the loss l_{after} is calculated on images from the mini-batch X^b. The loss change $(l_{after} - l_{before})$ accounts for the impact of X^a on X^b. For evaluating the impact of learning the clean images for the adversarial images $clean \rightarrow adv$, X^a are from the clean images and X^b are from the adversarial images.

* Corresponding Author

B Details of "Loss change validation"

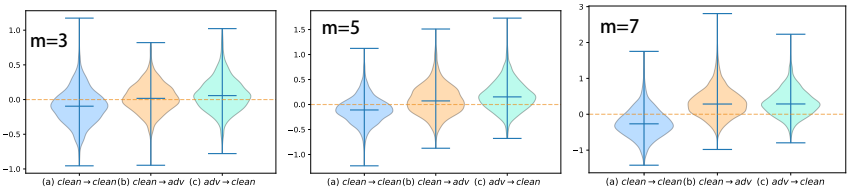

Fig. I: Loss change validation with different steps (m).

Fig. I shows the results of the MTD model under different values of the hyper-parameter "m". It can be noticed that the larger m is the greater the variation of the loss is. So we choose $m = 5$ in our experiments. These results are statistical results obtained from **3000 times of experiments**, with two batches of images randomly selected for each experiment. We attack both classification and localization loss in these experiments. These results capture the overall properties of the object detector. The adversarial images are generated by the model before tuning.

C Evaluation Results for Each Category

Table I: The performance comparison of standard SSD (non-robust), MTD, our RobustDet and RobustDet*, on clean images and adversarial images under two attacks (A_{cls} and A_{loc}), for each category.

Method	Clean				A_{cls}				A_{loc}			
	SSD	MTD [5]	RobustDet	RobustDet*	SSD	MTD [5]	RobustDet	RobustDet*	SSD	MTD [5]	RobustDet	RobustDet*
aeroplane	**81.4**	57.3	80.4	79.8	0.4	49.1	**64.2**	59.6	2.3	46.4	**61.5**	59.3
bicycle	**85.7**	66.7	83.0	83.9	0.9	34.8	48.9	**53.5**	6.0	44.2	54.0	**63.4**
bird	**75.3**	36.7	70.3	73.0	6.1	23.6	25.5	**34.8**	0.9	21.6	28.9	**37.8**
boat	**69.4**	28.7	66.3	67.2	0.1	14.8	**38.7**	27.4	0.6	17.3	**40.1**	31.4
bottle	**50.2**	19.7	46.0	46.4	0.8	10.6	14.1	**18.4**	9.2	10.3	16.8	**21.0**
bus	83.7	61.5	**84.0**	83.4	1.1	51.3	**58.6**	52.5	11.4	48.4	**62.4**	58.0
car	**85.4**	71.2	84.7	84.5	0.9	47.6	**60.6**	53.9	11.7	54.9	**62.6**	61.9
cat	**87.5**	48.1	85.2	84.3	0.1	36.0	44.7	**62.7**	9.3	23.4	48.9	**57.0**
chair	**61.6**	31.4	59.0	57.2	1.0	16.8	24.2	**31.5**	9.1	17.6	24.8	**27.9**
cow	**83.0**	37.9	79.9	78.0	0.1	11.9	21.8	**24.9**	3.8	15.5	33.8	**44.6**
diningtable	**79.4**	47.7	76.4	75.3	1.5	38.8	52.9	**65.7**	4.0	35.1	56.3	**61.7**
dog	**84.4**	49.3	82.4	82.0	0.3	29.1	43.7	**51.9**	9.1	31.8	41.2	**54.0**
horse	**86.1**	66.8	85.7	85.5	0.4	37.9	49.8	**62.5**	1.4	44.9	59.0	**65.8**
motorbike	84.2	62.4	83.2	**84.6**	0.9	28.9	47.9	**48.4**	1.5	44.2	59.3	**61.1**
person	**78.0**	57.7	77.1	77.1	5.0	41.1	50.7	**55.6**	9.6	42.5	51.6	**56.1**
pottedplant	**49.3**	20.8	46.5	46.5	0.1	4.0	**12.8**	11.9	3.1	9.7	16.4	**18.7**
sheep	**75.5**	32.5	73.3	72.5	0.1	10.9	15.7	**29.2**	1.9	18.3	34.1	**42.7**
sofa	**78.9**	58.8	75.3	75.7	0.3	51.6	56.6	**60.5**	2.3	43.6	**53.0**	51.7
train	**85.6**	62.7	84.3	84.8	1.3	39.0	49.4	**56.7**	1.5	39.7	55.8	**59.3**
tvmonitor	**75.3**	50.4	75.3	74.6	9.1	42.3	46.4	**56.4**	6.1	37.8	49.4	**50.8**

From the experimental results of each category, it indicates that the performance of MTD [5] on clean images has a large deviation from the standard SSD model [3], especially serious about the performance deficit of boat and bottle, with a loss of 40.7% and 30.5% mAP, respectively. The performance of our RobustDet on clean images is comparable to the standard SSD, and even exceeds the standard SSD for the category of bus. Under the A_{cls} attack, the performance of the standard SSD on adversarial images is inferior for all classes. The performance of MTD under this attack is quite limited and is even only 4.0% mAP on potted-plant. Instead, our RobustDet outperforms MTD for all classes on both A_{cls} and A_{loc} adversarial images. With the aid of CFR, RobustDet* further improves the detection performance for most object categories on both A_{cls} and A_{loc} adversarial images. The comparison results for each category with CWAT [1] are not included because those results are not published and source codes are not available.

D More Visualizations of Detection Results

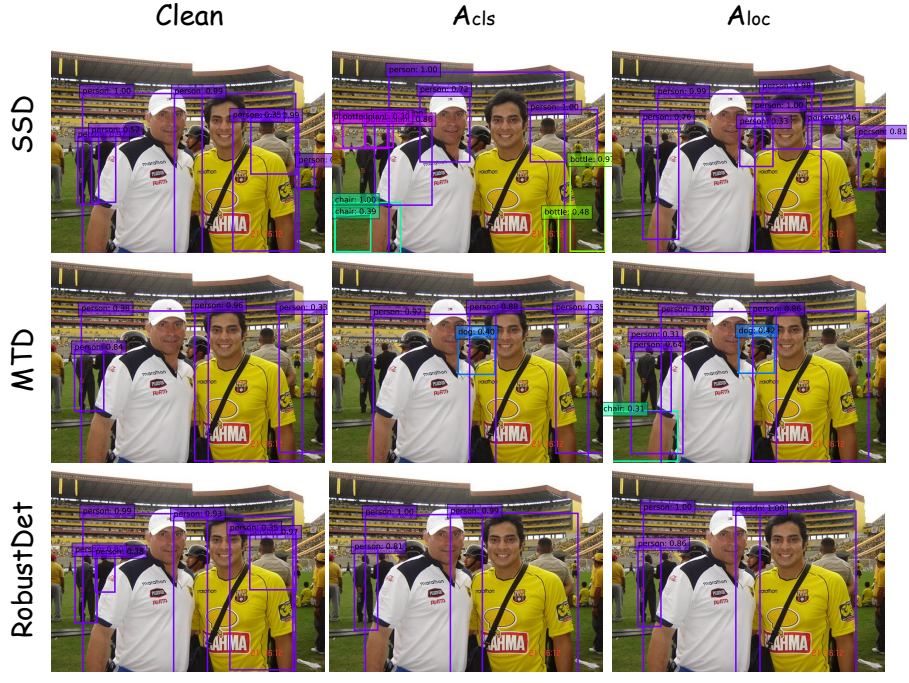

Fig. II: Visualization results of the non-robust SSD and robust models (*i.e.*, MTD and our RobustDet) on clean images and two different adversarial images.

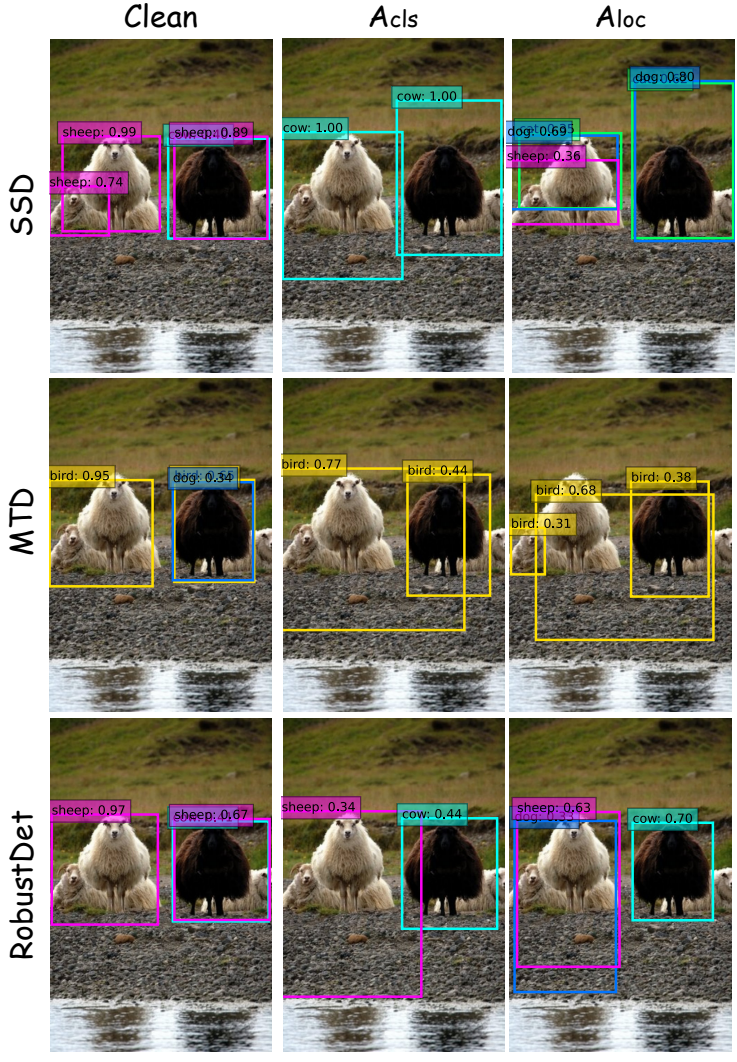

Fig. III: Visualization results of the non-robust SSD and robust models (*i.e.*, MTD and our RobustDet) on clean images and two different adversarial images.

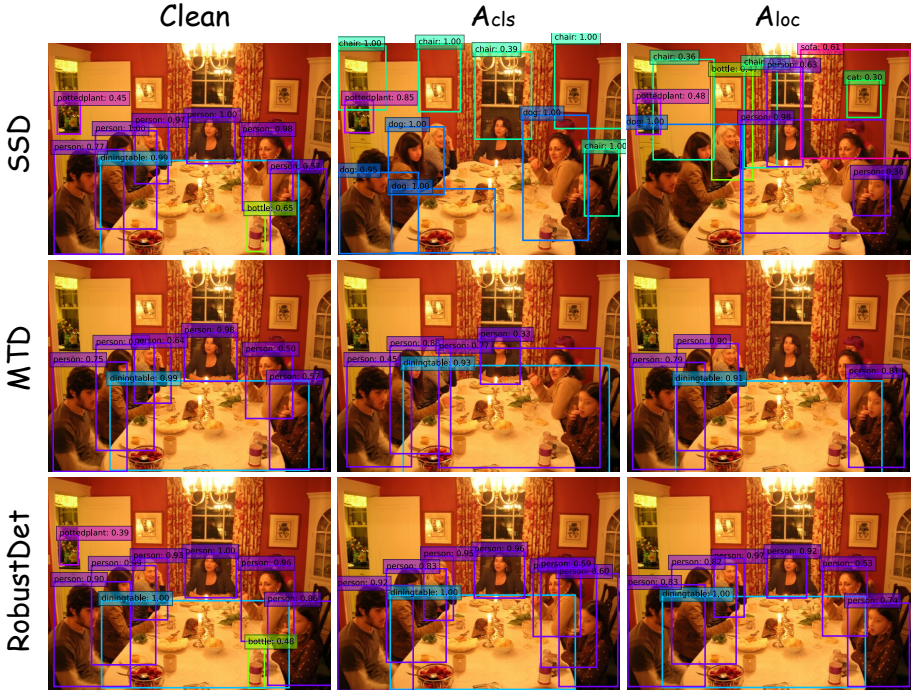

Fig. IV: Visualization results of the non-robust SSD and robust models (*i.e.*, MTD and our RobustDet) on clean images and two different adversarial images.

Fig. V: Visualization results of the non-robust SSD and robust models (*i.e.*, MTD and our RobustDet) on clean images and two different adversarial images.

Fig. VI: Visualization results of the non-robust SSD and robust models (*i.e.*, MTD and our RobustDet) on clean images and two different adversarial images.

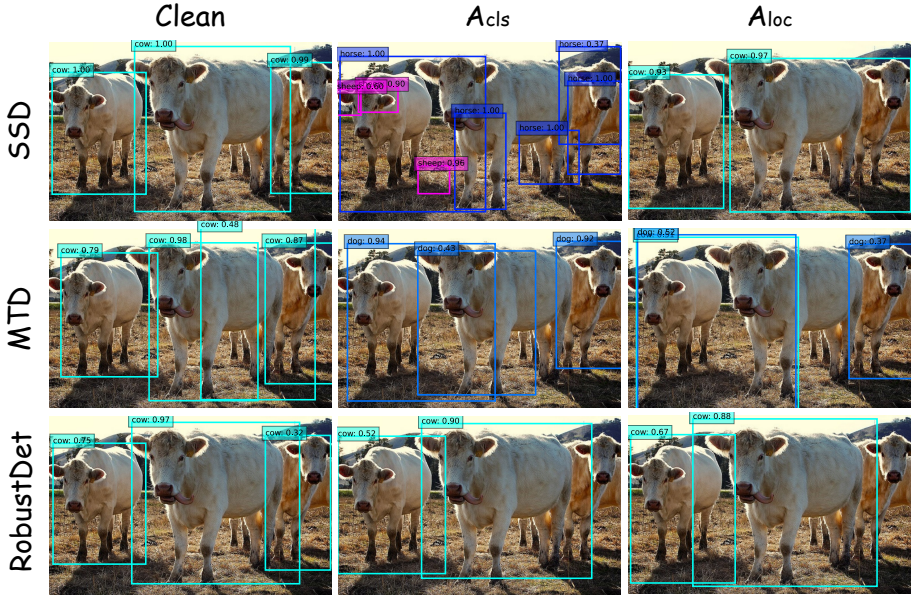

Fig. VII: Visualization results of the non-robust SSD and robust models (*i.e.*, MTD and our RobustDet) on clean images and two different adversarial images.

E Evaluating our method on more detector architectures

Besides SSD as a basemodel in the paper, we further evaluate our method with another two detectors, two-stage Faster RCNN [4] and anchor-free YOLOX [2] in Tab.II. It is observed that our RobustDet (implemented based on these two detectors) still significantly outperforms existing related method MTD and achieves comparable performance on clean images as standard SSD.

Table II: The **bold**/**bold** indicates the highest performance, and the blue indicates the highest performance among robust models.

Method	Faster-RCNN			YOLOX		
	clean	A_{cls}	A_{loc}	clean	A_{cls}	A_{loc}
SSD	**69.9**	1.9	1.12	**83.6**	2.8	5.5
MTD	53.7	8.8	16.1	64.2	18.1	25.2
RobustDet (ours)	67.9	**26.5**	**38.0**	79.5	**32.3**	**39.7**

F Running time and memory consumption

(1) *Efficiency:* Our method runs faster 2.5x to reach the model convergence with the similar loss error (*e.g.*, MTD costs 44h (hours) while our RobustDet costs only 18h).

(2) *Memory:* Our method has a slight memory consumption increase due to dynamic convolutions, compared with existing methods, MTD and CWAT. But even so, our RobustDet runs efficiently as discussed above for efficiency.

G The effectiveness of CFR

From the experimental results in Tab.2 of the main paper, it can be seen that RobusDet* performs not better than RobustDet on the MS-COCO dataset. Only one difference between these two model is RobustDet* has the CFR module. To figure out this issue on CFR, we have a further investigation by evaluating our RobustDet with or without (w/o) CFR on an MS-COCO subset, which has the same 20 object categories as PASCAL VOC. From results in Tab.III, the mAP performance of adding CFR is better after reducing the categories. This indicates that RobustDet with CFR perform better on adversarial images for the task with fewer object categories. To a certain extend, reconstruction can be treated as VAE in VGG-16 whose capacity is relatively limited to learn so many categories, thus compromising the overall training of the model and leading to the performance degradation.

Table III: The evaluation results of MS-COCO with the same 20 object categories as PASCAL VOC.

Method	clean	A_{cls}	A_{loc}
RobustDet (w/o CFR)	**13.3**	6.4	6.3
RobustDet	11.7	**7.2**	**6.9**

H Inject CFR into other layers

In the main paper, we inject CFR into the conv4_3 layer, but we also tried to insert it into other layers (conv5_3 and conv3_3). The experimental results in Tab.IV show that inserting CFR into the conv4_3 performs better.

Table IV: The evaluation results of inject CFR into other layers on PASCAL VOC.

Layer	clean	A_{cls}	A_{loc}	CWA
conv3_3	73.8	41.8	48.4	43.5
conv4_3	**74.8**	**45.9**	**49.1**	**48.0**
conv5_3	**74.8**	42.6	47.9	43.1

References

1. Chen, P., Kung, B., Chen, J.: Class-aware robust adversarial training for object detection. In: Computer Vision and Pattern Recognition (CVPR). pp. 10420–10429 (2021) 3
2. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430 (2021) 9
3. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C., Berg, A.C.: SSD: single shot multibox detector. In: European Conference on Computer Vision (ECCV). pp. 21–37 (2016) 3
4. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 91–99 (2015) 9
5. Zhang, H., Wang, J.: Towards adversarially robust object detection. In: International Conference on Computer Vision (ICCV). pp. 421–430 (2019) 2, 3