Enhancing Cross-task Black-Box Transferability of Adversarial Examples with Dispersion Reduction
Supplementary Material

Yantao Lu*
Syracuse University
ylu25@syr.edu

Yunhan Jia*
Bytedance AI Lab
yunhan.jia@bytedance.com

Jianyu Wang
Baidu USA
wjyouch@gmail.com

Bai Li
Duke University
bai.li@duke.edu

Weiheng Chai
Syracuse University
wchai01@syr.edu

Lawrence Carin
Duke University
lcarin@duke.edu

Senem Velipasalar
Syracuse University†
svelipas@syr.edu

A. Target models

The backbones and datasets of pretrained weights for target models are shown in Table 1.

Models	Backbone	Pretrained Dataset
Yolov3[11][10]	DarkNet53	COCO
RetineNet[4][2]	ResNet50	COCO
SSD[5][8]	MobileNet	COCO
Faster R-CNN[12][9]	ResNet50	COCO
Mask R-CNN[3][9]	ResNet50	COCO
DeepLabv3[1][9]	ResNet101	sub COCO in VOC labels
FCN [6][9]	ResNet101	sub COCO in VOC labels

Table 1: Backbone and pretrained dataset for target models.

B. Experiments on ImageNet

We have performed adversarial attacks on randomly chosen 5000 correctly classified images from the ImageNet validation set. The accuracies for detection and segmentation are shown in Table 3 and Table 4, respectively. Since there are no ground truth annotations and masks for the test images, the performance metrics are selected as the relative mAP/mIoU for detection and semantic segmentation respectively. In other words, the predictions from benign samples are regarded as the ground truth and predictions from adversarial examples are regarded as inference results.

Our proposed method (DR) achieves the best results in 17 out of 21 sets of experiments (81.0%) by degrading the performance of the target model by a larger margin. For detection, our proposed attack reduces the mAP, on average, to 7.41 over all the experiments. It creates 3.8 more drop in mAP compared to the best of the baselines (TI-DIM: 11.2 mAP). For semantic segmentation, our proposed attack achieves 16.93 mIoU on average over all the experiments. It achieves 4.76 more drop in mIoU compared to the best of the baselines (DIM: 21.69 mIoU).

Avg. Res.	Det. mAP	Seg. mIoU
	COCO&VOC/ImageNet	
PGD	26.1 / 19.1	33.6 / 28.8
MI-FGSM	22.8 / 15.6	30.6 / 25.2
DIM	18.6 / 11.5	25.9 / 21.8
TI-DIM	16.7 / 11.2	26.4 / 21.7
DR (Ours)	**12.8 / 7.4**	**20.0 / 16.9**

Table 2: Average results for detection and segmentation using COCO, VOC and ImageNet validation images.

C. Average Results

We have compared the proposed DR attack with the state-of-the-art adversarial techniques to demonstrate the transferability of our method on public object detection and semantic segmentation models. We have used the validation sets of ImageNet, VOC2012 and COCO for testing object detection and semantic segmentation tasks. The average results can be seen in Table 2,

For COCO and VOC datasets, our proposed method (**DR**) achieves the best results by degrading the performance of the target model by a larger margin. For detection, our proposed drops the mAP to 12.8 on average over all the experiments. It creates 3.9 more drop in mAP compared to the best of the baselines (TI-DIM: 16.7 mAP). For semantic

*Equal contribution

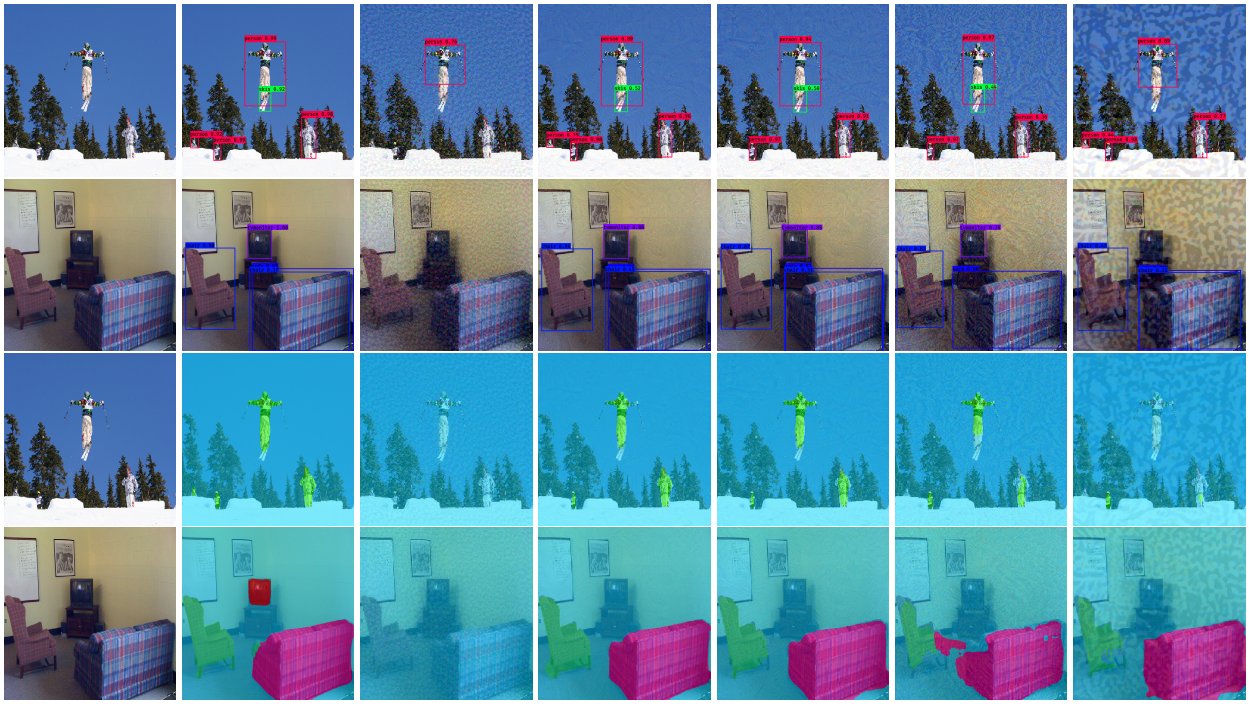
†The information, data, or work presented herein was funded in part by National Science Foundation (NSF) under Grant 1739748, Grant 1816732 and by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0000940. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

		Yolov3 DrkNet mAP	RetinaNet ResNet50 mAP	SSD MobileNet mAP	Faster-RCNN ResNet50 mAP	Mask-RCNN ResNet50 mAP
VGG16	PGD(α=1,N=20)	31.6	19.1	19.5	6.4	7.1
	PGD(α=4,N=100)	18.7	7.0	7.7	2.8	3.3
	MI-FGSM(α=1,N=20)	25.9	13.4	15.2	4.7	5.0
	MI-FGSM(α=4,N=100)	16.4	5.0	6.6	1.8	2.2
	DIM(α=1,N=20)	23.4	11.3	11.5	3.7	4.5
	DIM(α=4,N=100)	17.2	5.8	6.3	2.2	2.7
	TI-DIM(α=1.6,N=20)	21.5	10.2	11.6	3.5	4.0
	TI-DIM(α=4,N=100)	**16.3**	7.8	8.6	2.3	2.7
	DR(α=4,N=100)(**ours**)	17.0	**3.6**	**4.1**	**1.2**	**1.5**
InceptionV3	PGD(α=1,N=20)	51.3	36.6	33.9	25.9	25.1
	PGD(α=4,N=100)	33.3	16.4	16.2	14.1	14.7
	MI-FGSM(α=1,N=20)	44.6	27.4	27.5	19.8	20.1
	MI-FGSM(α=4,N=100)	30.3	14.1	15.3	11.9	12.5
	DIM(α=1,N=20)	30.6	15.2	16.4	11.0	11.7
	DIM(α=4,N=100)	25.3	10.2	10.6	6.9	8.2
	TI-DIM(α=1.6,N=20)	30.6	15.4	16.1	9.4	10.3
	TI-DIM(α=4,N=100)	23.7	11.2	12.2	6.8	7.0
	DR(α=4,N=100)(**ours**)	**21.1**	**8.6**	**9.4**	**4.5**	**5.3**
Resnet152	PGD(α=1,N=20)	40.8	27.6	27.0	10.4	10.8
	PGD(α=4,N=100)	27.2	13.4	13.0	5.0	6.1
	MI-FGSM(α=1,N=20)	33.9	20.3	21.2	7.6	8.0
	MI-FGSM(α=4,N=100)	24.6	11.4	11.8	3.9	4.7
	DIM(α=1,N=20)	26.9	13.2	13.0	4.4	5.3
	DIM(α=4,N=100)	22.2	9.3	8.7	2.9	3.7
	TI-DIM(α=1.6,N=20)	25.3	13.0	13.3	4.2	5.0
	TI-DIM(α=4,N=100)	**19.5**	9.4	9.8	2.7	2.9
	DR(α=4,N=100)(**ours**)	21.0	**6.2**	**4.8**	**1.3**	**1.6**

Table 3: Detection results for ImageNet.

		DeepLabv3 ResNet101 mIoU	FCN ResNet101 mIoU
VGG16	PGD(α=1,N=20)	30.3	24.6
	PGD(α=4,N=100)	17.5	15.1
	MI-FGSM(α=1,N=20)	25.4	20.8
	MI-FGSM(α=4,N=100)	**15.5**	13.9
	DIM(α=1,N=20)	24.7	19.0
	DIM(α=4,N=100)	17.1	14.5
	TI-DIM(α=1.6,N=20)	23.8	20.0
	TI-DIM(α=4,N=100)	18.3	16.5
	DR(α=4,N=100)(**ours**)	16.5	**12.4**
InceptionV3	PGD(α=1,N=20)	47.3	37.5
	PGD(α=4,N=100)	31.0	24.4
	MI-FGSM(α=1,N=20)	40.5	31.8
	MI-FGSM(α=4,N=100)	28.3	22.8
	DIM(α=1,N=20)	30.4	24.4
	DIM(α=4,N=100)	25.0	20.0
	TI-DIM(α=1.6,N=20)	28.1	24.4
	TI-DIM(α=4,N=100)	22.1	20.6
	DR(α=4,N=100)(**ours**)	**19.7**	**17.2**
Resnet152	PGD(α=1,N=20)	39.5	31.1
	PGD(α=4,N=100)	26.4	20.9
	MI-FGSM(α=1,N=20)	33.5	26.3
	MI-FGSM(α=4,N=100)	24.5	19.3
	DIM(α=1,N=20)	26.8	21.0
	DIM(α=4,N=100)	21.7	17.3
	TI-DIM(α=1.6,N=20)	26.2	21.9
	TI-DIM(α=4,N=100)	**20.1**	18.3
	DR(α=4,N=100)(**ours**)	20.5	**15.3**

Table 4: Segmentation Results for ImageNet.



(a) Clean Data (b) Benign (c) DR (ours) (d) PGD (e) MI-FGSM (f) DIM (g) TI-DIM

Figure 1: Samples of Detection and Segmentation Results

segmentation, our proposed attack causes the mIoU to drop to 20.0 on average over all the experiments. It achieves 5.9 more drop in mIoU compared to the best of the baselines (DIM: 25.9 mIoU).

The diagnostic of average results for ImageNet can be seen in B.

D. Visualization

D.1. Sample Images

Figure 1 shows the visualization samples for the proposed method and baselines attacks. Examples of detection and segmentation results for clean images, results for benign images, proposed DR images, PGD images, MI-FGSM images, DIM images and TI-DIM images are shown in each column (starting from left), respectively. First two rows are the detection results, and the last two rows are the segmentation results. We can see that the proposed DR attack is able to effectively perform vanishing attack to both segmentation and detection tasks. It is also noted that the proposed DR attack is more successful and effective, compared to the baselines, when attacking and degrading the performance for smaller objects.

D.2. Difference Images

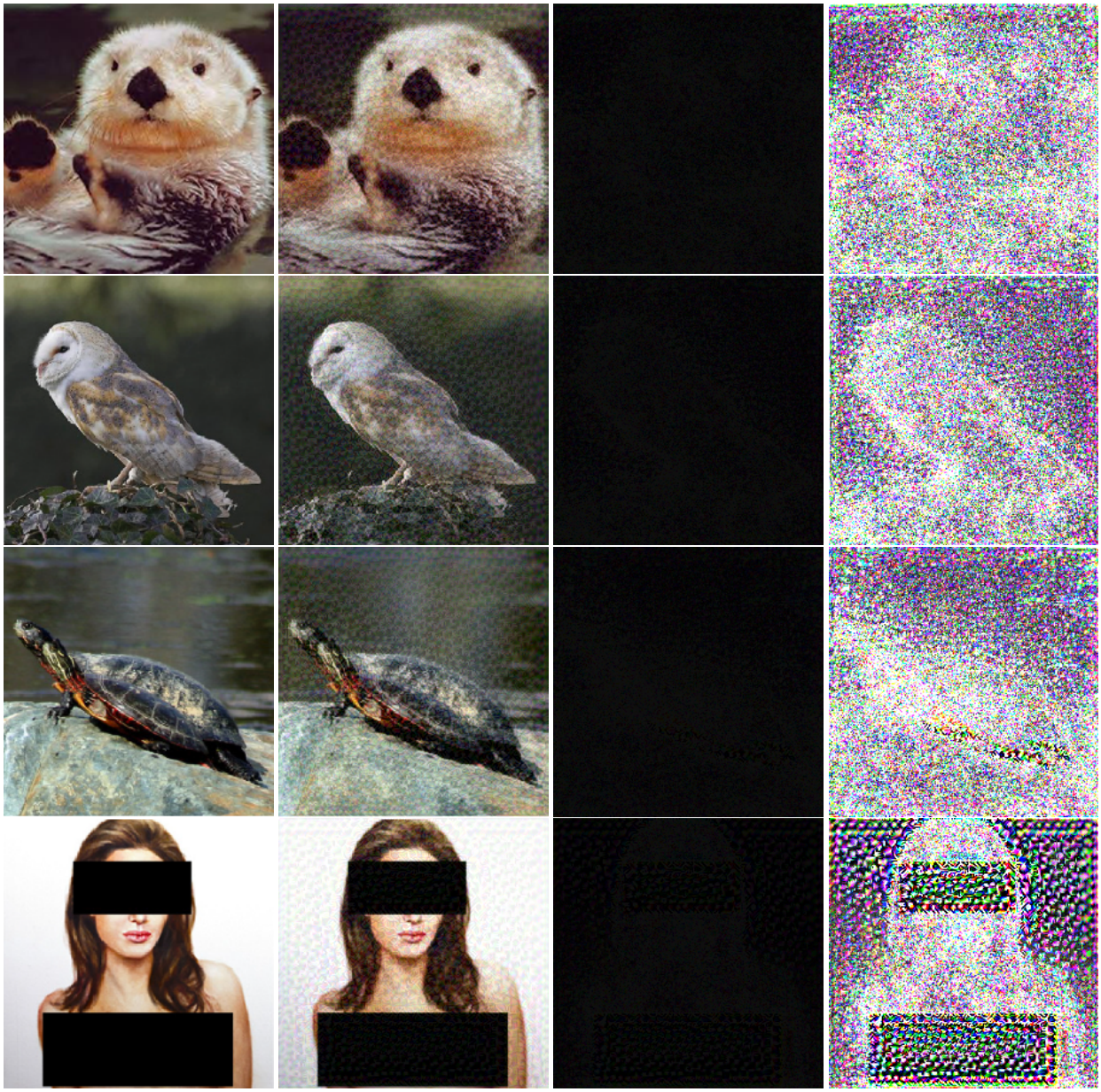
We implement VGG16 conv3.3 as the source model. The difference between original images and adversarial images are shown in Figure 2.

D.3. Perturbation Comparison

In Fig. 3, we present a sample of original image as well as AEs generated by baselines and our proposed DR method. As can be seen from the figure, the AE generated by our proposed method has the same perceptibly with which generated by the baseline methods.

E. Attacks Using Partial Feature Map

In this section, we show the experimental results for applying different dispersion reduction strategies. Pervasive standard deviation reduction, high value standard deviation reduction and masked standard deviation reduction are performed in the experiment. Pervasive standard deviation reduction is referred as classical std. reduction. For high value standard deviation reduction, in the feature output map, the elements that have highest 20% output values are gathered to perform std. reduction. For masked standard deviation reduction, a half height half width bounding box is implemented to locate at the area with highest output values. Elements within the bounding box are gathered to perform std. reduction. Then, the bounding box location are back traced to input image. The out-of-bbox perturbation are ignored while generating AEs. The retailed results are shown in Tab. 5 and a set of sample images are shown in Fig. 4. It



(a) Ori. (b) DR (c) Diff. (d) Diff.x15.94

Figure 2: **Samples of Difference Between Original Images and Adversarial Images.** VGG16 conv3.3 is chosen as the source model. The original images (a), adversarial images (b), difference images (c) and x15.94 amplified difference images (d) are shown in the figure by columns.

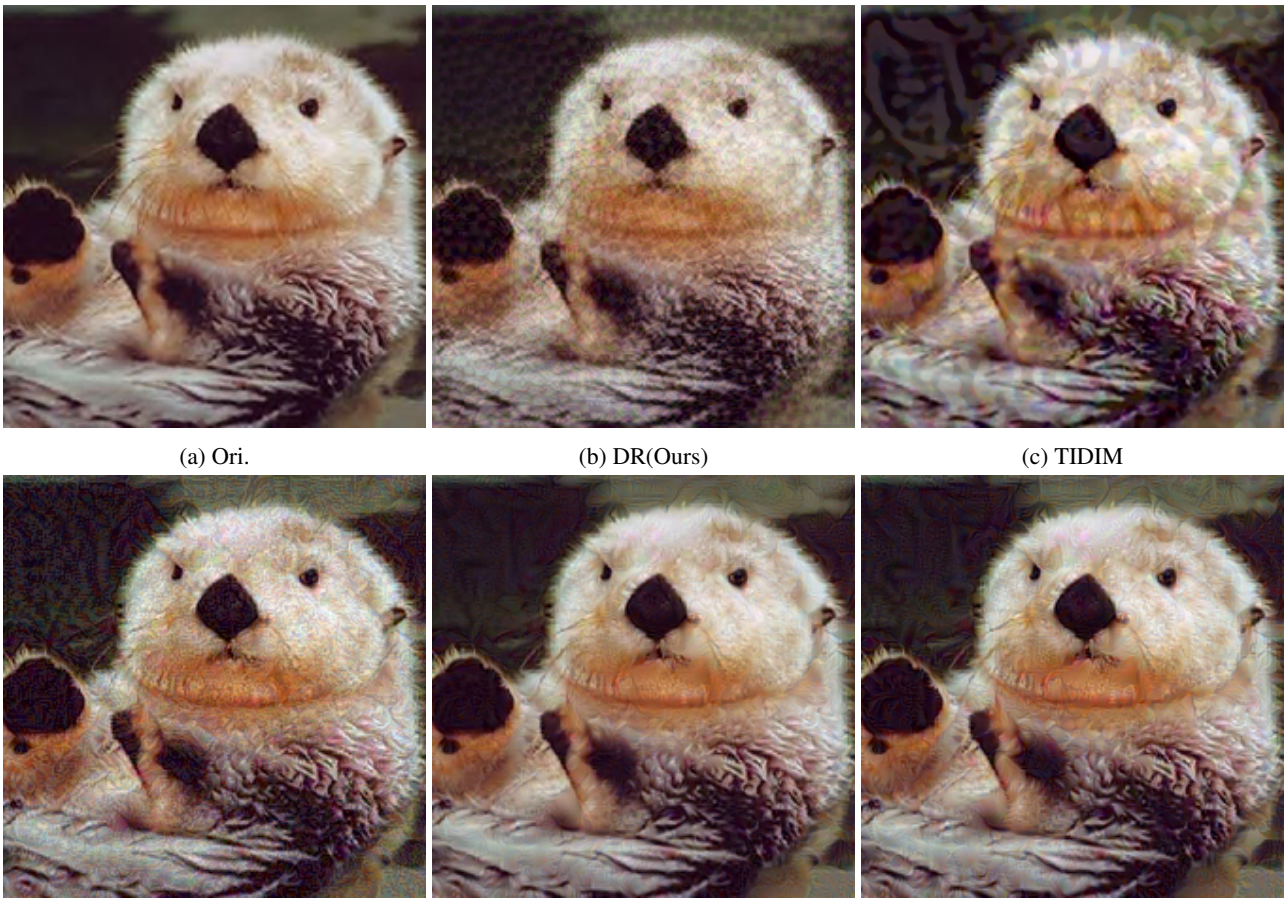
can be seen that the generated AEs that generated by modified std. reductions are more similar to the original image.

F. Attacking by $L_{inf} = 16$ and $L_{inf} = 8$

In this section, we evaluate performance for baselines and proposed DR method by AEs that are generated under $L_{inf} = 8$ restriction. COCO2017 are chosen as testing dataset and all attacking methods use $stepsize = 1$ and $numofsteps = 20$. We choose YOLOv3 and Deeplabv3 as target models to represent object detection and semantic segmentation. VGG16, InceptionV3 and Resnet152 are set as surrogates. The results are shown in Tab. 7. A set of sample images that compare $L_{inf} = 16$, $L_{inf} = 8$ and the original image are shown in Fig. 5.

G. Ensemble results on COCO2017

We compare the performance of proposed DR attack using single source models and ensemble model as surro-



(a) Ori. (b) DR(Ours) (c) TIDIM

(d) DIM (e) PGD (f) MIFGSM

Figure 3: **Compare Perturbations Generated by Baselines and Proposed Method.** The original images (a), AE generated by proposed DR (b), TI-DIM (c), DIM (d), PGD (e) and MI-FGSM (f) are shown in the figure, respectively. We can see that the AE generated by our proposed method has the same perceptibly with which generated by the baseline methods.

COCO2017 mAP/mIoU	DR step size=1 n_steps=20	yolov3 Drk	Retina Res50	FstrRCNN Res50	MaskRCNN Res50	Deeplabv3 Res101	FCN Res101
VGG16	pervasive	21.19	5.92	2.91	3.25	20.78	14.24
	selective	44.58	24.61	18.13	19.05	52.68	38.49
	mask	21.64	6.43	3.17	4.71	21.29	14.90
Incv3	pervasive	24.81	10.40	10.46	11.60	25.77	18.73
	selective	31.49	13.80	16.40	17.67	26.71	19.56
	mask	29.22	12.54	15.51	15.74	29.34	20.94
Res152	pervasive	24.30	8.29	3.13	4.12	26.66	18.60
	selective	43.65	23.43	17.57	18.23	51.62	37.51
	mask	27.51	9.25	3.60	6.16	28.43	20.94

Table 5: **Pervasive, selective and partial attack.** For high value standard deviation reduction, in the feature output map, the elements that have highest 20% output values are gathered to perform std. reduction. For masked standard deviation reduction, a half height half width bounding box is implemented to locate at the area with highest output values. Elements within the bounding box are gathered to perform std. reduction. Then, the bounding box location are back traced to input image.

gate. As shown in Tab. 6, ensemble attacking slightly outperforms single attacking. Ensemble attack achieves best performance in 3 over 6 sets of experiments and slightly out performs in other 3 experiments. Due to the trade-off of slightly performance improvement and significant computational expanse increment, we do not recommend using ensemble attacking as baseline for the proposed method.

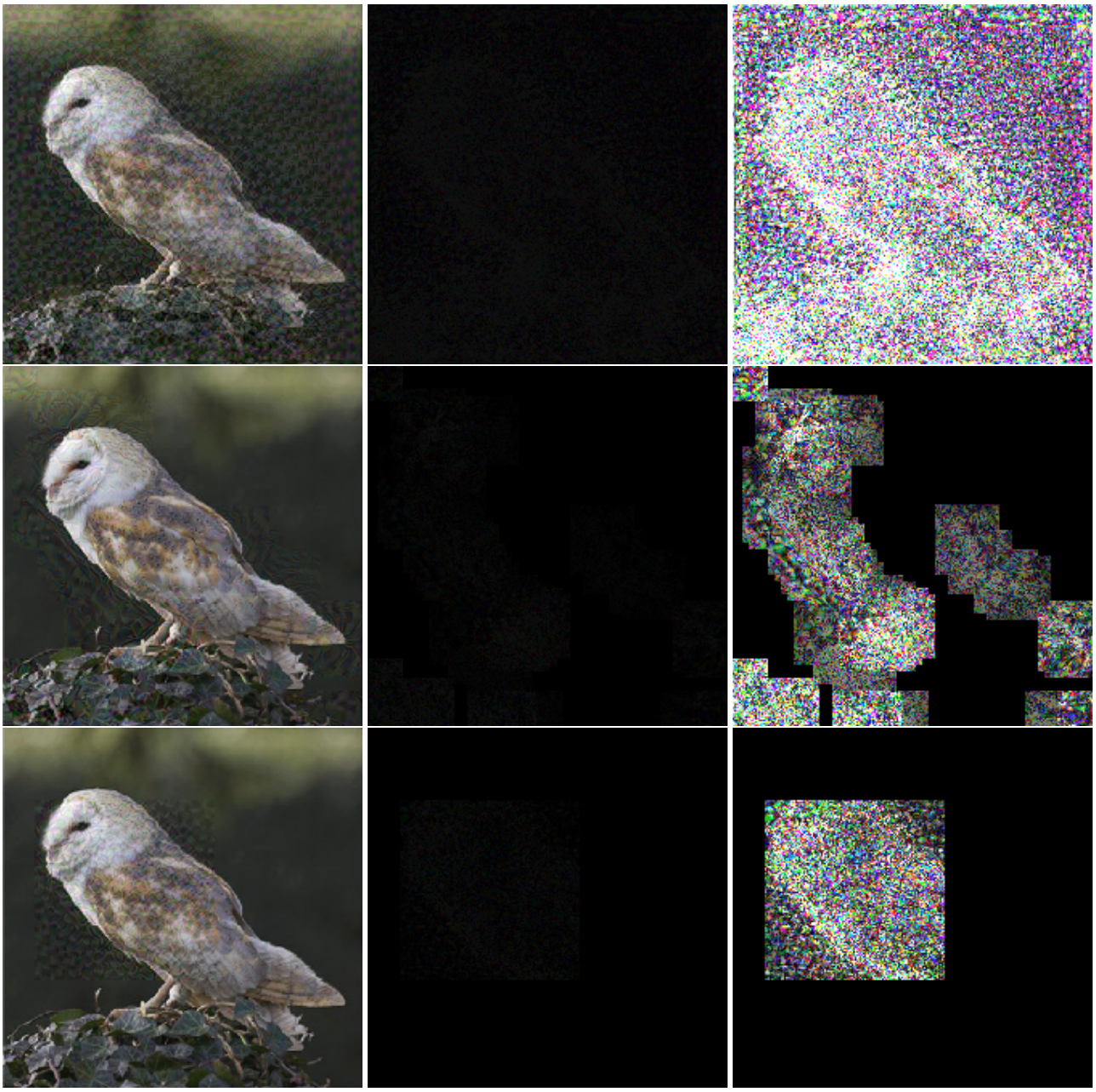


Figure 4: **Pervasive, selective and partial attack.** Pervasive standard deviation reduction, high value standard deviation reduction and masked standard deviation reduction are shown in the figure. It can be seen that the generated AEs that generated by modified std. reductions are more similar to the original image.

COCO 2017 mAP/mIoU	yolov3 Drk	Retina Res50	FstrRCNN Res50	MaskRCNN Res50	Deeplabv3 Res101	FCN Res101
VGG16	19.82	5.29	2.46	3.17	17.16	12.92
Incv3	24.18	8.52	8.30	9.79	23.23	17.05
Res152	22.72	6.83	2.25	3.03	22.66	16.35
ensemble	19.24	5.36	2.53	3.06	17.06	12.69

Table 6: **Results for comparing AEs generated by single source models and ensemble model.** Ensemble attack achieves best performance in 3 over 6 sets of experiments and slightly out performs in other 3 experiments.

(a) $L_{inf} = 8$ (b) $L_{inf} = 16$ (c) Original

Figure 5: **Perturbed Images Generated by Different Metrics.** Without contrasting to the original image, the perturbation generated by $L_{inf} = 8$ are imperceptible by humans [7].

COCO2017 mAP/mIoU	Eps8	yolov3 Drk	Deeplabv3 Res101
VGG16	MIFGSM	34.66	38.69
	PGD	37.33	42.04
	DIM	36.77	41.21
	TIDIM	37.29	43.51
	DR(Ours)	31.45	34.41
Incv3	MIFGSM	45.20	46.90
	PGD	40.08	49.66
	DIM	42.84	47.13
	TIDIM	44.09	48.79
	DR(Ours)	41.13	42.08
Res152	MIFGSM	39.51	43.96
	PGD	47.48	46.33
	DIM	37.26	42.50
	TIDIM	38.98	45.34
	DR(Ours)	33.58	36.52

Table 7: **Detection and Segmentation results for AEs generated by $L_{inf} = 8$ metric on COCO2017.** We can see that our proposed DR attack achieves better attacking performance comparing to SOTA baselines.

References

[1] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017. 1

[2] fizyr. keras retinanet. https://github.com/fizyr/keras-retinanet. 1

[3] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. 1

[4] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, Oct 2017. 1

[5] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. *CoRR*, abs/1512.02325, 2015. 1

[6] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014. 1

[7] Yan Luo, Xavier Boix, Gemma Roig, Tomaso A. Poggio, and Qi Zhao. Foveation-based mechanisms alleviate adversarial examples. *arXiv preprint arXiv:1511.06292*, 2015. 7

[8] pierluigiferrari. ssd keras. https://github.com/pierluigiferrari/ssd_keras. 1

[9] Pytorch. Torchvision models. https://pytorch.org/docs/master/torchvision/models.html. 1

[10] qqwweee. keras yolo3. https://github.com/qqwweee/keras-yolo3. 1

[11] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018. 1

[12] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015. 1