

DISCO: Adversarial Defense with Local Implicit Functions

Anonymous Author(s)

Affiliation

Address

email

1 A Compare to SOTA in RobustBench

2 In this section, the quantitative result of the baselines in RobustBench [14] are listed. Table A, C and
 3 D correspond to Fig.6(a), (b) and (c) of the main paper, respectively. Table B shows the baselines
 4 under Autoattack with $\epsilon_2 = 0.5$. The index in each table is associated to the index of Fig.6 in the
 5 main paper. The baselines in each table are categorized into No defense (first block), Adversarially
 6 trained defense in RobustBench (second block), Transformation based defense (third block) and
 7 DISCO (last block). The results of adversarially trained baselines are copied from RobustBench,
 8 while the results of transformation based defense are conducted by ourselves. For STL [60], different
 9 sparse constraints λ from the publicly available STL github¹ are used. DISCO is also combined with
 10 various classifiers for evaluation. More discussion can be found in Sec. 4.1 of the paper.

ID	Method	Standard Acc.	Robust Acc.	Avg. Acc.	Model	ID	Method	Standard Acc.	Robust Acc.	Avg. Acc.	Model
0	No Defense	94.78	0	47.39	WRN28-10	1	Rebuffi et al. [48]	92.23	66.58	79.41	WRN70-16
1	Rebuffi et al. [48]	92.23	66.58	79.41	WRN70-16	2	Gowal et al. [25]	88.74	66.11	77.43	WRN70-16
3	Gowal et al. [24]	91.1	63.88	78.49	WRN70-16	4	Rebuffi et al. [48]	88.5	64.64	76.57	WRN106-16
5	Rebuffi et al. [48]	88.54	64.25	76.4	WRN70-16	6	Kang et al. [34]	93.73	71.28	82.51	WRN70-16
7	Gowal et al. [25]	87.5	63.44	75.47	WRN28-10	8	Pang et al. [41]	89.01	63.35	76.18	WRN70-16
9	Rade et al. [47]	91.47	62.83	77.15	WRN34-10	10	Schwag et al. [53]	87.3	62.79	75.05	ResNest152
11	Gowal et al. [24]	89.48	62.8	76.14	WRN28-10	12	Huang et al. [27]	91.23	62.54	76.89	WRN34-R
13	Huang et al. [27]	90.56	61.56	76.06	WRN34-R	14	Dai et al. [18]	87.02	61.55	74.29	WRN28-10
15	Pang et al. [41]	88.61	61.04	74.83	WRN28-10	16	Rade et al. [47]	88.16	60.97	74.57	WRN28-10
17	Rebuffi et al. [48]	87.33	60.75	74.04	WRN28-10	18	Wu et al. [66]	87.67	60.65	74.16	WRN34-15
19	Sridhar et al. [59]	86.53	60.41	73.47	WRN34-15	20	Schwag et al. [54]	86.68	60.27	73.48	WRN34-10
21	Wu et al. [67]	88.25	60.04	74.15	WRN28-10	22	Schwag et al. [54]	89.46	59.66	74.56	WRN28-10
23	Zhang et al. [77]	89.36	59.64	74.5	WRN28-10	24	Yair et al. [8]	89.69	59.53	74.61	WRN28-10
25	Gowal et al. [25]	87.35	58.63	72.99	PreActRes18	26	Addepalli et al. [11]	85.32	58.04	71.68	WRN34-10
27	Chen et al. [10]	86.03	57.71	71.87	WRN34-20	28	Rade et al. [47]	89.02	57.67	73.35	PreActRes18
29	Gowal et al. [24]	85.29	57.2	71.25	WRN70-16	30	Schwag et al. [55]	88.98	57.14	73.06	WRN28-10
31	Rade et al. [47]	86.86	57.09	71.98	PreActRes18	32	Chen et al. [10]	85.21	56.94	71.08	WRN34-10
33	Gowal et al. [24]	85.64	56.86	71.25	WRN34-20	34	Rebuffi et al. [48]	83.53	56.66	70.1	PreActRes18
35	Wang et al. [63]	87.5	56.29	71.9	WRN28-10	36	Wu et al. [67]	85.36	56.17	70.77	WRN34-10
37	Alayrac et al. [3]	86.46	56.03	71.25	WRN28-10	38	Schwag et al. [54]	84.59	55.54	70.07	Res18
39	Dan et al. [26]	87.11	54.92	71.02	WRN28-10	40	Pang et al. [43]	86.43	54.39	70.41	WRN34-20
41	Pang et al. [44]	85.14	53.74	69.44	WRN34-20	42	Cui et al. [17]	88.7	53.57	71.14	WRN34-20
43	Zhang et al. [76]	84.52	53.51	69.02	WRN34-10	44	Rice et al. [49]	85.34	53.42	69.38	WRN34-20
45	Huang et al. [28]	83.48	53.34	68.41	WRN34-10	46	Zhang et al. [74]	84.92	53.08	69	WRN34-10
47	Cui et al. [16]	88.22	52.86	70.54	WRN34-10	48	Qin et al. [46]	86.28	52.84	69.56	WRN40-8
49	Chen et al. [12]	86.04	51.56	68.8	Res50	50	Chen et al. [11]	85.32	51.12	68.22	WRN34-10
51	Addepalli et al. [2]	80.24	51.06	65.65	Res18	52	Chawin et al. [58]	86.84	50.72	68.78	WRN34-10
53	Engstrom et al. [22]	87.03	49.25	68.14	Res50	54	Sinha et al. [57]	87.8	49.12	68.46	WRN34-10
55	Mao et al. [38]	86.21	47.41	66.81	WRN34-10	56	Zhang et al. [71]	87.2	44.83	66.02	WRN34-10
57	Madry et al. [36]	87.14	44.04	65.59	WRN34-10	58	Maksym et al. [4]	79.84	43.93	61.89	PreActRes18
59	Pang et al. [42]	80.89	43.48	62.19	Res32	60	Wong et al. [64]	83.34	43.21	63.28	PreActRes18
61	Shafahi et al. [56]	86.11	41.47	63.79	WRN34-10	62	Ding et al. [19]	84.36	41.44	62.9	WRN28-4
63	Souvik et al. [33]	87.32	40.41	63.87	Res18	64	Matan et al. [6]	81.3	40.22	60.76	Res18
65	Moosavi-Dezfooli et al. [39]	83.11	38.5	60.81	Res18	66	Zhang et al. [72]	89.98	36.64	63.31	WRN28-10
67	Zhang et al. [73]	90.25	38.45	63.35	WRN28-10	68	Jang et al. [29]	78.91	34.95	56.93	Res20
69	Kim et al. [32]	91.51	34.22	62.87	WRN34-10	70	Zhang et al. [75]	44.73	32.64	38.69	5 layer CNN
71	Wang et al. [62]	92.8	29.35	61.08	WRN28-10	72	Xiao et al. [68]	79.28	18.5	48.89	DenseNet121
73	Jin et al. [30]	90.84	1.35	46.1	Res18	74	Aamir et al. [40]	89.16	0.28	44.72	Res110
75	Chan et al. [9]	93.79	0.26	47.03	WRN34-10						
76	Bit Reduction [70]	92.66	1.04	46.85	WRN28-10	77	Jpeg [21]	83.9	50.73	67.32	WRN28-10
78	Input Rand. [69]	94.3	8.59	51.45	WRN28-10	79	LIIF [13]	94.85	0.22	47.54	WRN28-10
80	AutoEncoder	76.54	67.41	71.98	WRN28-10	81	STL [60] (k=64 s=8 $\lambda=0.1$)	90.65	57.28	73.97	WRN28-10
82	STL [60] (k=64 s=8 $\lambda=0.15$)	86.77	66.94	76.86	WRN28-10	83	STL [60] (k=64 s=8 $\lambda=0.2$)	82.22	67.92	75.07	WRN28-10
84	Median Filter	79.67	42.49	61.08	WRN28-10						
85	DISCO	89.26	85.56 \pm 0.02	87.41	WRN28-10						

Table A: Cifar10 baselines and DISCO under Autoattack ($\epsilon_\infty = 8/255$). This table corresponds to Fig. 6(a) in the main paper.

¹https://github.com/GitBoSun/AdvDefense_CSC

ID	Method	Standard Acc.	Robust Acc.	Avg. Acc.	Model	ID	Method	Standard Acc.	Robust Acc.	Avg. Acc.	Model
0	No Defense	94.78	0	47.39	WRN28-10						
1	Rebuffi et al. [48]	95.74	82.32	89.03	WRN70-16	2	Gowal et al. [24]	94.74	80.53	87.64	WRN70-16
3	Rebuffi et al. [48]	92.41	80.42	86.42	WRN70-16	4	Rebuffi et al. [48]	91.79	78.8	85.30	WRN28-10
5	Augustin et al. [7]	93.96	78.79	86.38	WRN34-10	6	Schwag et al. [53]	90.93	77.24	84.09	WRN34-10
7	Augustin et al. [7]	92.23	76.25	84.24	WRN34-10	8	Rade et al. [47]	90.57	76.15	83.36	PreActRes18
9	Rebuffi et al. [48]	90.33	75.86	83.10	PreActRes18	10	Gowal et al. [24]	90.9	74.5	82.70	WRN70-16
11	Schwag et al. [53]	89.76	74.41	82.09	Res18	12	Wu et al. [67]	88.51	73.66	81.09	WRN34-10
13	Augustin et al. [7]	91.08	72.91	82.00	Res50	14	Engstrom et al. [22]	90.83	69.24	80.04	Res50
15	Rice et al. [49]	88.67	67.68	78.18	PreActRes18	16	Rony et al. [50]	89.05	66.44	77.75	WRN28-10
17	Ding et al. [19]	88.02	66.09	77.06	WRN28-4						
18	Bit Reduction [70]	92.66	3.8	48.23	WRN28-10	19	Jpeg [21]	83.9	69.85	76.88	WRN28-10
20	Input Rand. [69]	94.3	25.71	60.01	WRN28-10	21	LIIF [13]	94.85	0.22	47.54	WRN28-10
22	AutoEncoder	76.54	71.71	74.13	WRN28-10	23	STL [60] (k=64 s=8 $\lambda=0.1$)	90.65	75.55	83.1	WRN28-10
24	STL [60] (k=64 s=8 $\lambda=0.15$)	86.77	76.45	81.61	WRN28-10	25	STL [60] (k=64 s=8 $\lambda=0.2$)	82.22	74.33	78.28	WRN28-10
26	Median Filter	79.67	63.94	71.81	WRN28-10						
27	DISCO	89.26	88.47 \pm 0.16	88.87	WRN28-10						

Table B: Cifar10 baselines and DISCO under Autoattack ($\epsilon_2 = 0.5$).

ID	Method	Standard Acc.	Robust Acc.	Avg. Acc.	Model	ID	Method	Standard Acc.	Robust Acc.	Avg. Acc.	Model
0	No Defense	80.37	0	41.78	WRN28-10						
1	Gowal et al. [24]	69.15	36.88	53.02	WRN70-16	2	Rebuffi et al. [48]	63.56	34.64	49.1	WRN70-16
3	Pang et al. [41]	65.56	33.05	49.31	WRN70-16	4	Rebuffi et al. [48]	62.41	32.06	47.24	WRN28-10
5	Schwag et al. [53]	65.93	31.15	48.54	WRN34-10	6	Pang et al. [41]	63.66	31.08	47.37	WRN28-10
7	Chen et al. [10]	64.07	30.59	47.33	WRN34-10	8	Addepalli et al. [2]	65.73	30.35	48.04	WRN34-10
9	Cui et al. [17]	62.55	30.2	46.38	WRN34-20	10	Gowal et al. [24]	60.86	30.03	45.45	WRN70-16
11	Cui et al. [17]	60.64	29.33	44.99	WRN34-10	12	Rade et al. [47]	61.5	28.88	45.19	PreActRes18
13	Wu et al. [67]	60.38	28.86	44.62	WRN34-10	14	Rebuffi et al. [48]	56.87	28.5	42.69	PreActRes18
15	Dan et al. [26]	59.23	28.42	43.83	WRN28-10	16	Cui et al. [17]	70.25	27.16	48.71	WRN34-10
17	Addepalli et al. [2]	62.02	27.14	44.58	PreActRes18	18	Chen et al. [11]	62.15	26.94	44.55	WRN34-10
19	Chawin et al. [58]	62.82	24.57	43.7	WRN34-10	20	Rice et al. [49]	53.83	18.95	36.39	PreActRes18
21	Bit Reduction [70]	76.86	3.78	40.32	WRN28-10	22	Jpeg [21]	61.89	39.59	50.74	WRN28-10
23	Input Rand. [69]	73.57	3.31	38.44	WRN28-10	24	LIIF [13]	80.3	3.36	41.83	WRN28-10
25	AutoEncoder	58.79	48.36	53.575	WRN28-10	26	STL [60] (k=64 s=8 $\lambda=0.1$)	74.28	30.05	52.17	WRN28-10
27	STL [60] (k=64 s=8 $\lambda=0.15$)	70.3	41.82	56.06	WRN28-10	28	STL [60] (k=64 s=8 $\lambda=0.2$)	67.41	46.07	56.74	WRN28-10
29	Median Filter	65.78	34.52	50.15	WRN28-10						
30	DISCO	72.07	67.93 \pm 0.17	70	WRN28-10	31	DISCO	71.62	69.01 \pm 0.19	70.32	WRN34-10

Table C: Cifar100 baselines and DISCO under Autoattack ($\epsilon_\infty = 8/255$). This table corresponds to Fig. 6(b) in the main paper.

ID	Method	Standard Acc.	Robust Acc.	Avg. Acc.	Model	ID	Method	Standard Acc.	Robust Acc.	Avg. Acc.	Model
0	No Defense	76.52	0	38.26	Res50						
1	Hadi et al. [51]	68.46	38.14	53.3	WRN50-2	2	Hadi et al. [51]	64.02	34.96	49.49	Res50
3	Engstrom et al. [22]	62.56	29.22	45.89	Res50	4	Wong et al. [64]	55.62	26.24	40.93	Res50
5	Hadi et al. [51]	52.92	25.32	39.12	Res18						
6	Bit Reduction [70]	67.64	4.04	35.84	Res18	7	Bit Reduction [70]	73.82	1.86	37.84	Res50
8	Bit Reduction [70]	75.06	4.96	40.01	WRN50-2	9	Jpeg [21]	67.18	13.08	40.13	Res18
10	Jpeg [21]	73.64	33.42	53.53	Res50	11	Jpeg [21]	75.42	24.9	50.16	WRN50-2
12	Input Rand. [69]	64	17.78	40.89	Res18	13	Input Rand. [69]	74.02	18.84	46.43	Res50
14	Input Rand. [69]	71.7	23.58	47.64	WRN50-2	15	STL [60] (k=64 s=8 $\lambda=0.1$)	67.5	18.5	43	Res18
16	STL [60] (k=64 s=8 $\lambda=0.2$)	65.64	32.9	49.27	Res18	17	STL [60] (k=64 s=8 $\lambda=0.1$)	72.56	32.7	52.63	Res50
18	STL [60] (k=64 s=8 $\lambda=0.2$)	68.3	50.16	59.23	Res50	19	Median Filter	66.1	10.34	38.22	Res18
20	Median Filter	71.68	17.36	44.52	Res50						
21	DISCO	67.98	60.88 \pm 0.17	64.43	Res18	22	DISCO	72.64	68.2 \pm 0.29	70.42	Res50
23	DISCO	75.1	69.5 \pm 0.23	72.3	WRN50-2						

Table D: ImageNet baselines and DISCO under Autoattack ($\epsilon_\infty = 4/255$). This table corresponds to Fig. 6(c) in the main paper.

11 B Defense Transfer

12 In this section, the qualitative results of DISCO transferability across attacks are studied. Table E, F
 13 and G represents the results for Cifar10, Cifar100 and ImageNet, respectively. The corresponding
 14 plots are illustrated in Fig. A, B and C. More discussion can be found in Sec. 4.1 of the paper.

Table E: Defense Transfer across L_∞ attacks ($\epsilon_\infty = 8/255$) on Cifar10.

Method Classifier	Rebuffi et al. [48] WRN70-16	Gowal et al. [24] WRN28-10	DISCO WRN28-10
FGSM [23]	75.66	70.91	64.08
PGD [37]	69.93	66.02	82.99
BIM [34]	69.84	65.95	80.46
RFGSM [61]	69.8	65.95	81.2
EotPgd [35]	70.68	66.58	76.84
TPgd [74]	82.32	80.48	81.61
FFgsm [65]	78.04	73.37	70.1
MiFgsm [20]	73.22	68.82	45.49
APgd [15]	69.46	65.69	85.79
Jitter [52]	70.15	64.84	80.49
Avg.	72.72	68.69	75.88

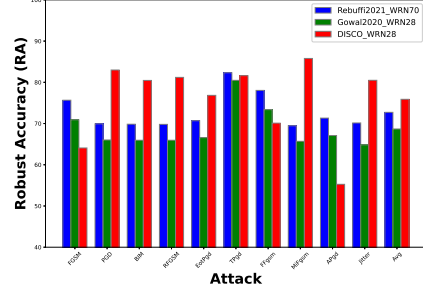


Figure A: Defense Transfer across L_∞ attacks on Cifar10.

Table F: Defense Transfer across L_∞ attacks ($\epsilon_\infty = 8/255$) on Cifar100.

Method Classifier	Gowal et al. [24] WRN70-16	Rebuffi et al. [48] WRN28-10	DISCO WRN28-10
FGSM [23]	44.53	38.57	50.4
PGD [37]	40.46	36.09	74.51
BIM [34]	40.38	36.03	72.25
RFGSM [61]	40.42	35.99	72.1
EotPgd [35]	41.07	36.45	74.8
TPgd [74]	57.52	52.01	74.06
FFgsm [65]	47.61	41.47	64.29
MiFgsm [20]	42.37	37.31	44.14
APgd [15]	39.99	35.64	77.33
Jitter [52]	38.38	33.04	73.75
Avg.	43.27	38.26	67.76

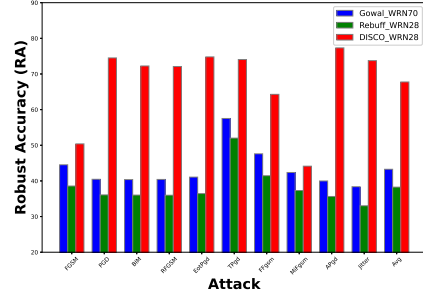


Figure B: Defense Transfer across L_∞ attacks on Cifar100.

Table G: Defense Transfer across L_∞ attacks ($\epsilon_\infty = 4/255$) on ImageNet.

Method Classifier	Hadi et al. [51] Res50	Engstrom et al. [22] Res50	DISCO Res50
Clean	64.1	62.54	72.64
FGSM [23]	43.48	39.96	55.72
PGD [37]	39.28	33.32	66.32
BIM [34]	39.26	33.2	66.4
RFGSM [61]	39.28	33.16	66.4
EotPgd [35]	41.2	37.24	69.32
TPgd [74]	53.82	49.64	69.94
FFgsm [65]	43.58	40.1	57
MiFgsm [20]	40.56	35.6	52.38
APgd [15]	38.42	32.22	68.3
Jitter [52]	36.26	31.36	67.04
Avg.	41.51	36.58	63.88

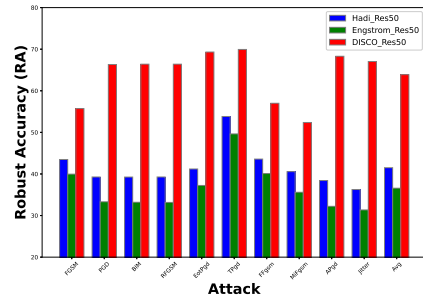


Figure C: Defense Transfer across L_∞ attacks on ImageNet.

15 C Improving Cifar10 and Cifar100 SOTA on RobustBench

16 Sec. 4.1 in the main paper shows that DISCO can improve the prior SOTA defenses on ImageNet
 17 dataset. In Table H, we further investigate the gain of applying DISCO on SOTA Cifar10 and Cifar100
 18 defense. The first and second block of Table H shows the gain after applying DISCO on [48], which

Table H: Improving SOTA defenses on RobustBench [14] for Cifar10 (L_2 and L_∞) and Cifar100 (L_∞) dataset.

Method	Dataset	Norm	SA	RA	Avg.
Rebuffi et al. [48]	Cifar10	L_∞	92.23	66.58	79.41
w/ DISCO	Cifar10	L_∞	91.95	70.71	81.33
Rebuffi et al. [48]	Cifar10	L_2	95.74	82.32	89.03
w/ DISCO	Cifar10	L_2	95.24	84.15	89.7
Gowal et al. [24]	Cifar100	L_∞	69.15	36.88	53.02
w/ DISCO	Cifar10	L_∞	68.56	39.77	54.17

19 is the prior SOTA defense on Cifar10 against L_2 and L_∞ Autoattack. DISCO also improves the
 20 prior SOTA defense [24] on Cifar100 by 2.89%. These results indicate that DISCO can not only be a
 21 robust defense by itself, but also applicable to other existing defense to improve their robustness.

22 D Kernel Size s

Table I: Ablation on various kernel size s .

s	SA	RA	Avg.
1	71.22	69.52	70.37
3	72.64	68.2	70.42
5	74.22	60.1	67.16

23 In this section, we ablate the kernel size used to train DISCO on ImageNet. The kernel size s
 24 controls the feature neighborhood forwarded to the local implicit module. Table I shows that $s = 3$
 25 achieves the best performance, which degrades for $s = 5$ by a significant margin (3.26%). This
 26 shows that while tasks like classification require large and global receptive fields, the projection of
 27 adversarial images into the natural image manifold can be done on small neighborhoods. Given that
 28 the complexity of modeling the manifold increases with the neighborhood size, it is not surprising
 29 that larger s lead to weaker performance. This is consistent with the well known complexity of
 30 synthesizing images with global models, such as GANs. What is somewhat surprising is that even
 31 $s = 1$ is sufficient to enable a robust defense. By default, we use $s = 3$ in all our experiments.

32 E Computation Time for STL and DISCO

Table J: The computation time between STL [60] and DISCO when consuming different input size. Note that the STL requires $36.34\times$ inference time when the image size increases from 32 to 224.

Dataset	Image Size	STL [60]	DISCO				
			(K=1)	(K=2)	(K=3)	(K=4)	(K=5)
Cifar10	32	0.65	0.011	0.021	0.031	0.037	0.048
ImageNet	224	23.71	0.027	0.081	0.134	0.191	0.251
Time Increase		$\times 36.34$	$\times 2.41$	$\times 3.86$	$\times 4.35$	$\times 5.14$	$\times 5.19$

33 Table J compares the inference time of STL [60], DISCO and the cascade DISCO (from $K = 2$
 34 to 5) on Cifar10 and ImageNet. For a single image of size 32×32 in Cifar10 dataset, STL requires
 35 $5.9\times$ (0.65 vs 0.011) inference time than DISCO ($K=1$). When the cascade DISCO is used, the
 36 computation time increases approximately in linear, as the K increases.

37 When a single image of size 224 in ImageNet dataset is considered, STL requires 23.71 second to
 38 process the image, while DISCO ($K=1$) only requires 0.027. The computation time difference on
 39 ImageNet increases to $878.15\times$ (23.71 vs 0.027), which is significantly larger than that of Cifar10
 40 ($5.9\times$). This indicates that DISCO is a better defense that suits for various input input size with minor
 41 additional computing cost.

42 F Training Details

43 For Cifar10 and Cifar100, we train the DISCO for 40 epochs, while for ImageNet, DISCO is trained
44 for 3 epochs since the images of ImageNet is larger and produce more random crops during training.
45 The learning rate is set to 0.0001 and the Adam optimizer is used for all experiments. All the
46 experiments are conducted using Pytorch [45]. All the time measurement of the baselines and DISCO
47 are conducted on a single Nvidia Titan Xp GPU with Intel Xeon CPU E5-2630 with batch size 1 and
48 averaged over 100 images.

49 G Adopted Code and Benchmark

50 In this section, we list the url links that are used for training and evaluating DISCO. To create the
51 adversarial-clean training pairs, we adopt the code from TorchAttack² and Ares³, which support
52 the multiple attack methods. These attack methods are then used to attack pretrained classifiers
53 of Cifar10, Cifar100 and ImageNet. We adopt the ResNet18 classifiers of Cifar10 from Ares³,
54 WideResNet classifiers of Cifar100 from this repository⁴ and the ResNet18 classifiers of ImageNet
55 from Pytorch [45].

56 To evaluate DISCO, we adopt the Autoattack from RobustBench [14]⁵ and compare to the pretrained
57 defenses on RobustBench leaderboard. In addition to Autoattack, we use AdverTorch⁶ library to
58 implement the BPDA [5] attack and TorchAttack⁷ library for other attacks, like FGSM [23] and
59 BIM [34].

60 For the adversarially trained defense baselines, we adopt the pretrained weights from Robust-
61 Bench [14]⁸, while the code of transformation based baselines are adopted from Ares³, Cifar
62 autoencoder⁹ and STL¹ [60]. For DISCO implementation, we adopt the code from LIIF¹⁰ [13].

63 H DCT Analysis

64 The effectiveness of perturbation removal can be analyzed in the frequency domain using Discrete
65 Cosine Transform (DCT). Given an input image x , the corresponding clean image can be denoted
66 as x_{cln} . The normalized difference (ND) after DCT over M input images and its associated clean
67 images $\mathcal{X}_{cln} = \{x_{cln}^i\}_{i=1}^M$ can be defined as

$$ND(\mathcal{X}) = \log\left(\frac{1}{M} \sum_{i=1}^M \left| \frac{DCT(x^i) - DCT(x_{cln}^i)}{DCT(x_{cln}^i)} \right| \right), \quad (1)$$

68 where \mathcal{X} can be an adversarial images \mathcal{X}_{Adv} , the outputs of defense model \mathcal{X}_{Def} or the outputs of
69 cascade defense model \mathcal{X}_{CDef} . Fig. D(a), (b) and (c) are the ND results of \mathcal{X}_{Adv} , \mathcal{X}_{Def} and \mathcal{X}_{CDef}
70 over 100 images ($M = 100$). Darker area indicates that there exists higher similarity between the
71 clean images and input images at specific frequency. Take Fig. D(a) for example. The dark area
72 concentrates at low frequency area (upper left corner), while the bright area on the lower right corner
73 shows that the adversarial noise are mostly high frequency. After the adversarial image is forwarded
74 through DISCO, Fig. D(b) shows the high frequency area becomes darker. Fig. D(d) further highlights
75 the difference between Fig. D(a) and (b) by illustrating $\mathcal{U}(ND(\mathcal{X}_{Adv}) - ND(\mathcal{X}_{Def}))$, where \mathcal{U} is a
76 unit step function. The white area in Fig. D(d) indicates that most of the high frequency perturbation
77 are removed from the adversarial images. Similarly, Fig. D(e) demonstrates that the cascade DISCO
78 further removes the high frequency perturbation when comparing Fig. D(b) and (c).

²<https://adversarial-attacks-pytorch.readthedocs.io/en/latest/>

³<https://github.com/thu-ml/ares>

⁴<https://github.com/xternalz/WideResNet-pytorch>

⁵<https://github.com/RobustBench/robustbench>

⁶<https://github.com/BorealisAI/advertorch>

⁷<https://adversarial-attacks-pytorch.readthedocs.io/en/latest/>

⁸<https://github.com/RobustBench/robustbench>

⁹<https://github.com/chenjie/PyTorch-CIFAR-10-autoencoder>

¹⁰<https://github.com/yinboc/liif>

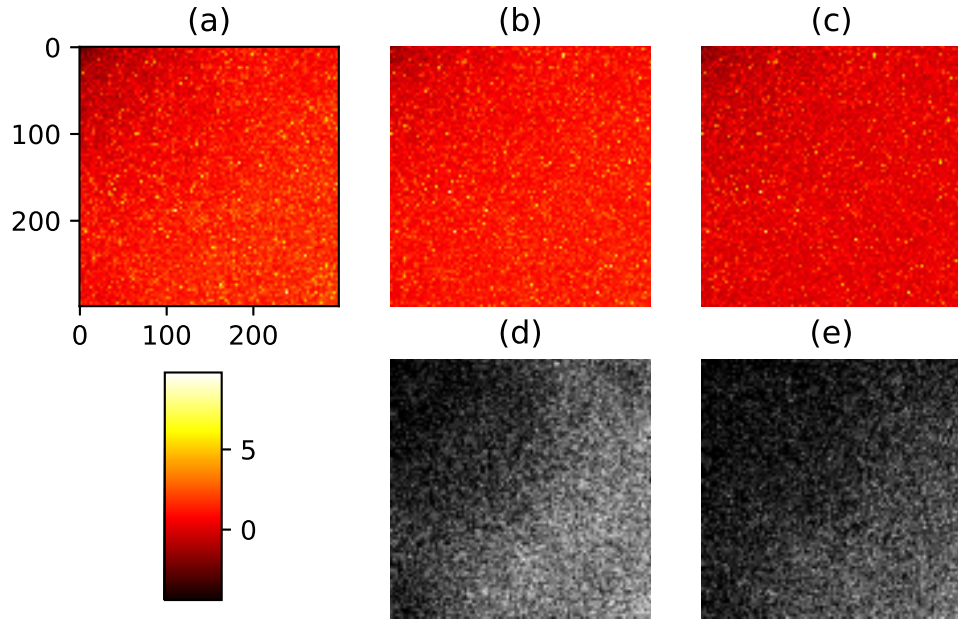


Figure D: (a), (b) and (c) are the ND results of \mathcal{X}_{Adv} , \mathcal{X}_{Def} and \mathcal{X}_{CDef} . (d) and (e) highlights the difference between (a)/(b) and (b)/(c). See text for more details.

79 I Visualizations

80 The disco defense outputs against FGSM [23], BIM [34] and PGD [37] attacks are visualized in
 81 Fig. E, F and G, respectively. Take Fig. E for example. The first and second row shows the clean
 82 and adversarial images. For 3-5 rows of Fig. E, the output of DISCO and cascade DISCO ($K = 2$
 83 and $K = 3$) are shown. Clearly, both DISCO and its cascade version can effectively remove the
 84 adversarial perturbation. In addition, Fig. H shows the DISCO output of various size from 128×128
 85 to 512×512 . Note that these images are produced from the same DISCO model without retraining for
 86 any output size or attack.

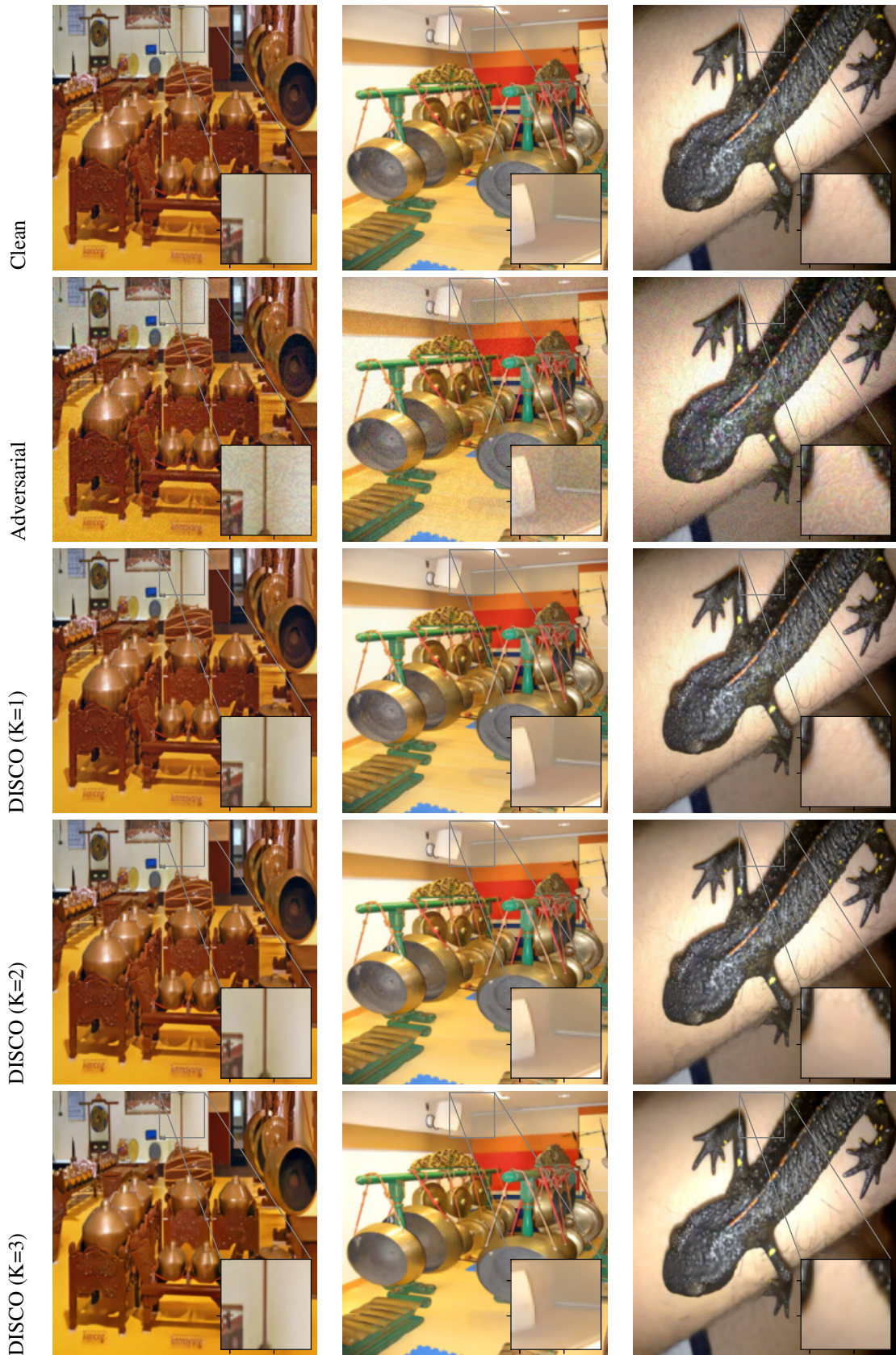


Figure E: Comparison of Clean image, Adversarial image and DISCO output from $K = 1$ to 3 under FGSM attack.

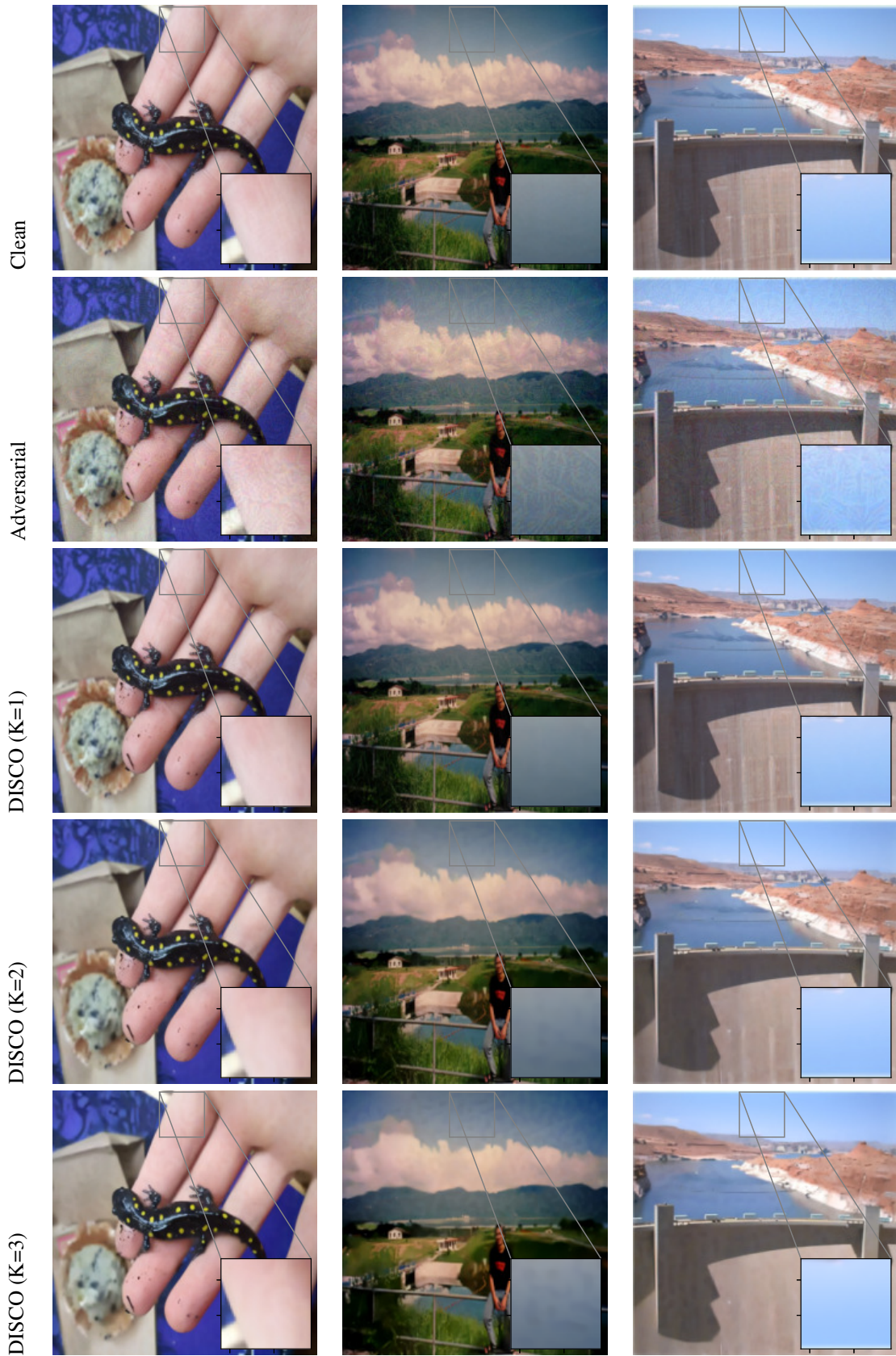


Figure F: Comparison of Clean image, Adversarial image and DISCO output from $K = 1$ to 3 under BIM attack.

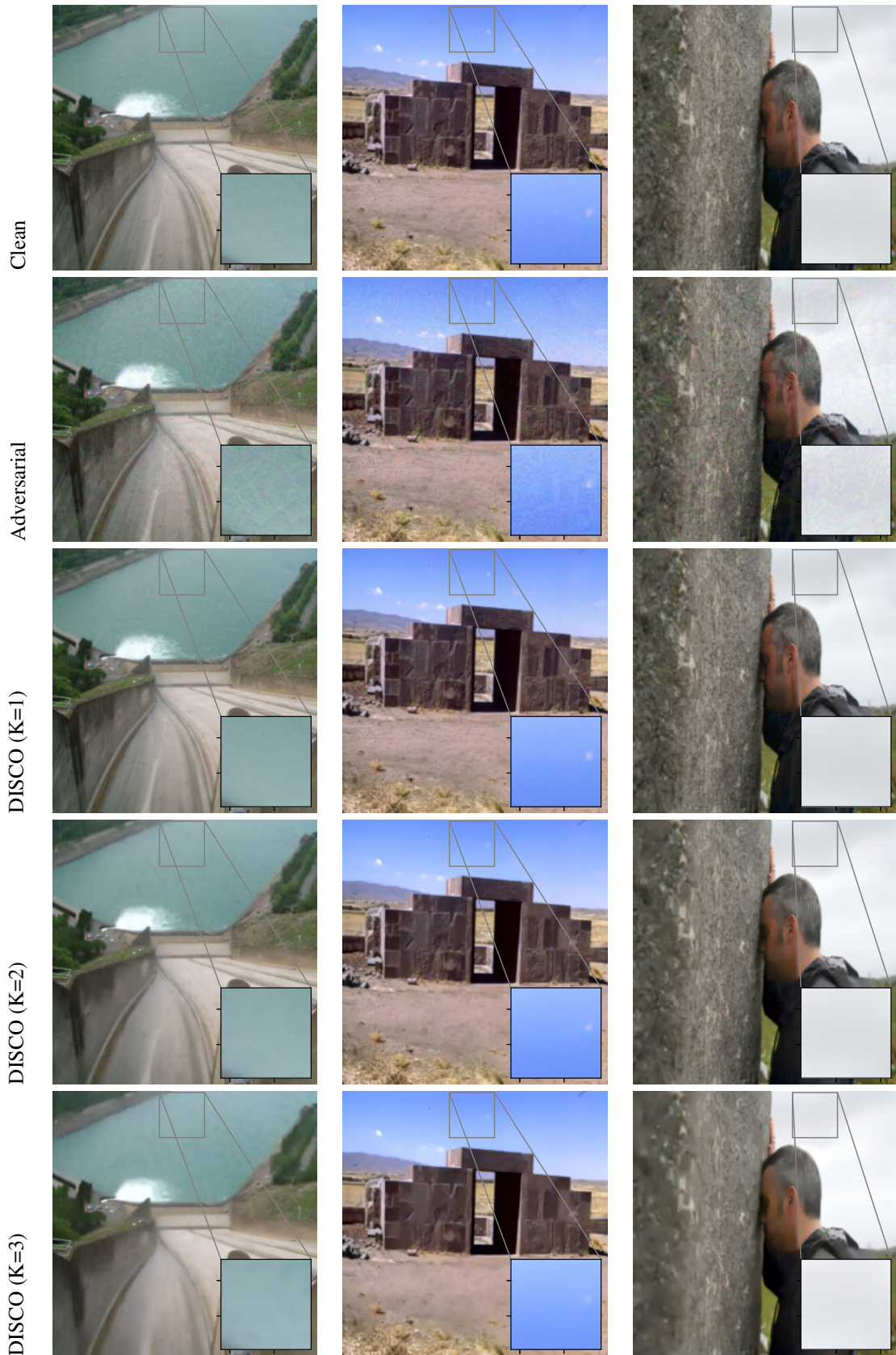
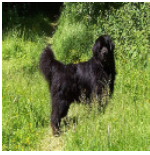


Figure G: Comparison of Clean image, Adversarial image and DISCO output from $K = 1$ to 3 under PGD attack.

Size: 128



Size: 256



Size: 512

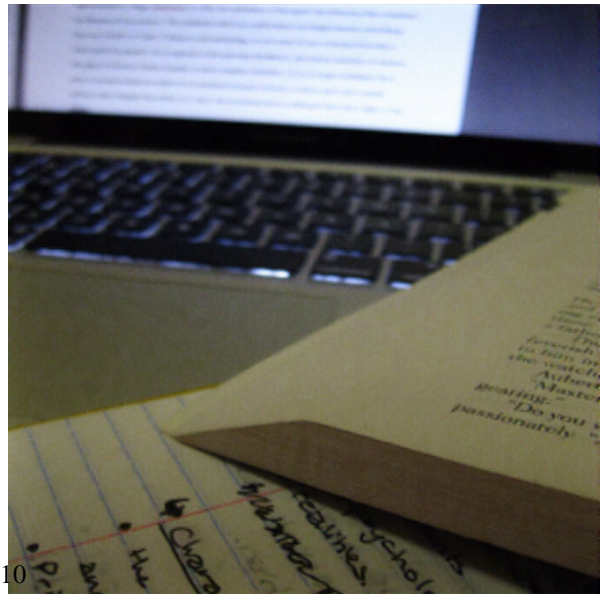
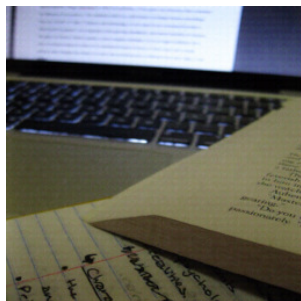
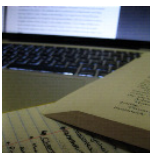
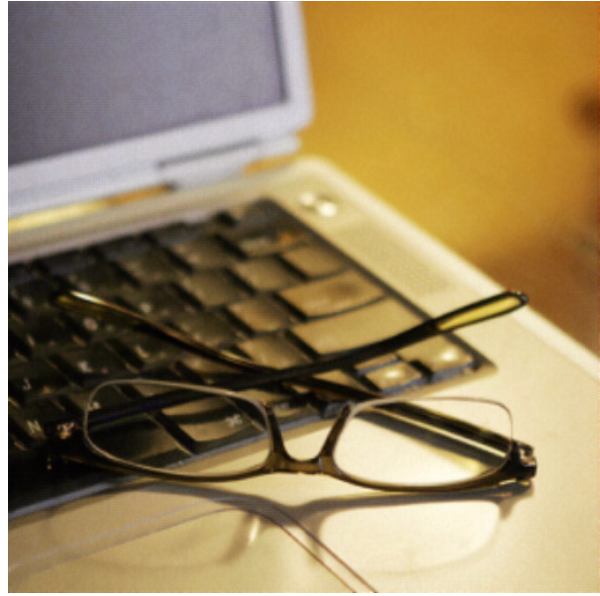


Figure H: Multiple output sizes (128, 256 and 512) of DISCO without re-training.

87 **Checklist**

- 88 1. For all authors...
- 89 (a) Do the main claims made in the abstract and introduction accurately reflect the paper's
90 contributions and scope? [Yes]
- 91 (b) Did you describe the limitations of your work? [Yes]
- 92 (c) Did you discuss any potential negative societal impacts of your work? [Yes]
- 93 (d) Have you read the ethics review guidelines and ensured that your paper conforms to
94 them? [Yes]
- 95 2. If you are including theoretical results...
- 96 (a) Did you state the full set of assumptions of all theoretical results? [N/A]
- 97 (b) Did you include complete proofs of all theoretical results? [N/A]
- 98 3. If you ran experiments...
- 99 (a) Did you include the code, data, and instructions needed to reproduce the main experi-
100 mental results (either in the supplemental material or as a URL)? [Yes] The code will
101 be released upon acceptance.
- 102 (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they
103 were chosen)? [Yes]
- 104 (c) Did you report error bars (e.g., with respect to the random seed after running experi-
105 ments multiple times)? [Yes]
- 106 (d) Did you include the total amount of compute and the type of resources used (e.g., type
107 of GPUs, internal cluster, or cloud provider)? [Yes]
- 108 4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
- 109 (a) If your work uses existing assets, did you cite the creators? [Yes]
- 110 (b) Did you mention the license of the assets? [N/A]
- 111 (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
112
- 113 (d) Did you discuss whether and how consent was obtained from people whose data you're
114 using/curating? [N/A]
- 115 (e) Did you discuss whether the data you are using/curating contains personally identifiable
116 information or offensive content? [N/A]
- 117 5. If you used crowdsourcing or conducted research with human subjects...
- 118 (a) Did you include the full text of instructions given to participants and screenshots, if
119 applicable? [N/A]
- 120 (b) Did you describe any potential participant risks, with links to Institutional Review
121 Board (IRB) approvals, if applicable? [N/A]
- 122 (c) Did you include the estimated hourly wage paid to participants and the total amount
123 spent on participant compensation? [N/A]

References

- [1] Sravanti Addepalli, Samyak Jain, Gaurang Sriramanan, Shivangi Khare, and Venkatesh Babu Radhakrishnan. Towards achieving adversarial robustness beyond perceptual limits. In *ICML 2021 Workshop on Adversarial Machine Learning*, 2021.
- [2] Sravanti Addepalli, Samyak Jain, Gaurang Sriramanan, Shivangi Khare, and Venkatesh Babu Radhakrishnan. Towards achieving adversarial robustness beyond perceptual limits. In *ICML 2021 Workshop on Adversarial Machine Learning*, 2021.
- [3] Jean-Baptiste Alayrac, Jonathan Uesato, Po-Sen Huang, Alhussein Fawzi, Robert Stanforth, and Pushmeet Kohli. Are labels required for improving adversarial robustness? In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.
- [4] Maksym Andriushchenko and Nicolas Flammarion. Understanding and improving fast adversarial training. In *NeurIPS*, 2020.
- [5] Anish Athalye, Nicholas Carlini, and David A. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, pages 274–283, 2018.
- [6] Matan Atzmon, Niv Haim, Lior Yariv, Ofer Israelov, Haggai Maron, and Yaron Lipman. Controlling neural level sets. In *Advances in Neural Information Processing Systems*, pages 2032–2041, 2019.
- [7] Maximilian Augustin, Alexander Meinke, and Matthias Hein. Adversarial robustness on in- and out-distribution improves explainability. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 228–245, Cham, 2020. Springer International Publishing.
- [8] Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. Unlabeled data improves adversarial robustness. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.
- [9] Alvin Chan, Yi Tay, Yew Soon Ong, and Jie Fu. Jacobian adversarially regularized networks for robustness. In *International Conference on Learning Representations*, 2020.
- [10] Erh-Chung Chen and Che-Rung Lee. Ltd: Low temperature distillation for robust adversarial training, 2021.
- [11] Jinghui Chen, Yu Cheng, Zhe Gan, Quanquan Gu, and Jingjing Liu. Efficient robust training via backward smoothing, 2021.
- [12] Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- [13] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8628–8638, 2021.
- [14] Francesco Croce, Maksym Andriushchenko, Vikash Sehwal, Edoardo DeBenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. RobustBench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.
- [15] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.
- [16] Jiequan Cui, Shu Liu, Liwei Wang, and Jiaya Jia. Learnable boundary guided adversarial training. *arXiv preprint arXiv:2011.11164*, 2020.

- 170 [17] Jiequan Cui, Shu Liu, Liwei Wang, and Jiaya Jia. Learnable boundary guided adversarial
171 training. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15701–
172 15710, 2021.
- 173 [18] Sihui Dai, Saeed Mahloujifar, and Prateek Mittal. Parameterizing activation functions for
174 adversarial robustness. *CoRR*, abs/2110.05626, 2021.
- 175 [19] Gavin Weiguang Ding, Yash Sharma, Kry Yik Chau Lui, and Ruitong Huang. Mma training: Di-
176 rect input space margin maximization through adversarial training. In *International Conference*
177 *on Learning Representations*, 2020.
- 178 [20] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li.
179 Boosting adversarial attacks with momentum. *2018 IEEE/CVF Conference on Computer Vision*
180 *and Pattern Recognition*, pages 9185–9193, 2018.
- 181 [21] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M. Roy. A study of the effect of
182 jpg compression on adversarial images. *ArXiv*, abs/1608.00853, 2016.
- 183 [22] Logan Engstrom, Andrew Ilyas, Hadi Salman, Shibani Santurkar, and Dimitris Tsipras. Robust-
184 ness (python library), 2019.
- 185 [23] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial
186 examples. In *International Conference on Learning Representations*, 2015.
- 187 [24] Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering
188 the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint*
189 *arXiv:2010.03593*, 2020.
- 190 [25] Sven Gowal, Sylvestre-Alvise Rebuffi, Olivia Wiles, Florian Stimberg, Dan Andrei Calian, and
191 Timothy A Mann. Improving robustness using generated data. In M. Ranzato, A. Beygelzimer,
192 Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information*
193 *Processing Systems*, volume 34, pages 4218–4233. Curran Associates, Inc., 2021.
- 194 [26] Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model
195 robustness and uncertainty. *Proceedings of the International Conference on Machine Learning*,
196 2019.
- 197 [27] Hanxun Huang, Yisen Wang, Sarah Monazam Erfani, Quanquan Gu, James Bailey, and Xingjun
198 Ma. Exploring architectural ingredients of adversarially robust deep neural networks. In
199 *NeurIPS*, 2021.
- 200 [28] Lang Huang, Chao Zhang, and Hongyang Zhang. Self-adaptive training: beyond empirical risk
201 minimization. In *Advances in Neural Information Processing Systems*, volume 33, 2020.
- 202 [29] Yunseok Jang, Tianchen Zhao, Seunghoon Hong, and Honglak Lee. Adversarial defense via
203 learning to generate diverse attacks. In *2019 IEEE/CVF International Conference on Computer*
204 *Vision (ICCV)*, pages 2740–2749, 2019.
- 205 [30] Charles Jin and Martin Rinard. Manifold regularization for locally stable deep neural networks,
206 2021.
- 207 [31] QIYU KANG, Yang Song, Qinxu Ding, and Wee Peng Tay. Stable neural ODE with
208 lyapunov-stable equilibrium points for defending against adversarial attacks. In A. Beygelz-
209 imer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information*
210 *Processing Systems*, 2021.
- 211 [32] Jungeum Kim and Xiao Wang. Sensible adversarial learning, 2020.
- 212 [33] Souvik Kundu, Mahdi Nazemi, Peter A. Beerel, and Massoud Pedram. Dnr: A tunable robust
213 pruning framework through dynamic network rewiring of dnns. In *Proceedings of the 26th Asia*
214 *and South Pacific Design Automation Conference, ASPDAC '21*, page 344–350, New York, NY,
215 USA, 2021. Association for Computing Machinery.
- 216 [34] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical
217 world. *ICLR Workshop*, 2017.

- 218 [35] Xuanqing Liu, Yao Li, Chongruo Wu, and Cho-Jui Hsieh. Adv-BNN: Improved adversarial
219 defense through robust bayesian neural network. In *International Conference on Learning*
220 *Representations*, 2019.
- 221 [36] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu.
222 Towards deep learning models resistant to adversarial attacks. In *International Conference on*
223 *Learning Representations*, 2018.
- 224 [37] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu.
225 Towards deep learning models resistant to adversarial attacks. In *International Conference on*
226 *Learning Representations*, 2018.
- 227 [38] Chengzhi Mao, Ziyuan Zhong, Junfeng Yang, Carl Vondrick, and Baishakhi Ray. Metric
228 learning for adversarial robustness. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-
229 Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*,
230 volume 32. Curran Associates, Inc., 2019.
- 231 [39] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard.
232 Robustness via curvature regularization, and vice versa. *2019 IEEE/CVF Conference on*
233 *Computer Vision and Pattern Recognition (CVPR)*, pages 9070–9078, 2019.
- 234 [40] Aamir Mustafa, Salman Khan, Munawar Hayat, Roland Goecke, Jianbing Shen, and Ling Shao.
235 Adversarial defense by restricting the hidden space of deep neural networks. In *The IEEE*
236 *International Conference on Computer Vision (ICCV)*, October 2019.
- 237 [41] Tianyu Pang, Min Lin, Xiao Yang, Jun Zhu, and Shuicheng Yan. Robustness and accuracy
238 could be reconcilable by (proper) definition, 2022.
- 239 [42] Tianyu Pang, Kun Xu, Yinpeng Dong, Chao Du, Ning Chen, and Jun Zhu. Rethinking
240 softmax cross-entropy loss for adversarial robustness. In *International Conference on Learning*
241 *Representations*, 2020.
- 242 [43] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial
243 training. In *International Conference on Learning Representations*, 2021.
- 244 [44] Tianyu Pang, Xiao Yang, Yinpeng Dong, Kun Xu, Jun Zhu, and Hang Su. Boosting adversarial
245 training with hypersphere embedding. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan,
246 and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages
247 7779–7792. Curran Associates, Inc., 2020.
- 248 [45] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan,
249 Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas
250 Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy,
251 Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-
252 performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-
253 Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*,
254 pages 8024–8035. Curran Associates, Inc., 2019.
- 255 [46] Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhus-
256 sein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial robustness through
257 local linearization. In *NeurIPS*, 2019.
- 258 [47] Rahul Rade and Seyed-Mohsen Moosavi-Dezfooli. Helper-based adversarial training: Reducing
259 excessive margin to achieve a better accuracy vs. robustness trade-off. In *ICML 2021 Workshop*
260 *on Adversarial Machine Learning*, 2021.
- 261 [48] Sylvestre-Alvise Rebuffi, Sven Gowal, Dan Andrei Calian, Florian Stimberg, Olivia Wiles,
262 and Timothy A. Mann. Fixing data augmentation to improve adversarial robustness. *ArXiv*,
263 abs/2103.01946, 2021.
- 264 [49] Leslie Rice, Eric Wong, and J. Zico Kolter. Overfitting in adversarially robust deep learning. In
265 *ICML*, 2020.

- 266 [50] Jérôme Rony, Luiz G. Hafemann, Luiz Oliveira, Ismail Ben Ayed, Robert Sabourin, and Eric
267 Granger. Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and
268 defenses. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*,
269 pages 4317–4325, 2019.
- 270 [51] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do
271 adversarially robust imagenet models transfer better? In H. Larochelle, M. Ranzato, R. Had-
272 sell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*,
273 volume 33, pages 3533–3545. Curran Associates, Inc., 2020.
- 274 [52] Leo Schwinn, René Raab, An Nguyen, Dario Zanca, and Bjoern M. Eskofier. Exploring robust
275 misclassifications of neural networks to enhance adversarial attacks. 2021.
- 276 [53] Vikash Sehwal, Saeed Mahloujifar, Tinashe Handina, Sihui Dai, Chong Xiang, Mung Chiang,
277 and Prateek Mittal. Robust learning meets generative models: Can proxy distributions improve
278 adversarial robustness? In *International Conference on Learning Representations*, 2022.
- 279 [54] Vikash Sehwal, Saeed Mahloujifar, Tinashe Handina, Sihui Dai, Chong Xiang, Mung Chiang,
280 and Prateek Mittal. Robust learning meets generative models: Can proxy distributions improve
281 adversarial robustness? In *International Conference on Learning Representations*, 2022.
- 282 [55] Vikash Sehwal, Shiqi Wang, Prateek Mittal, and Suman Jana. Hydra: Pruning adversarially
283 robust neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.
- 284 [56] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John P. Dickerson, Christoph Studer,
285 Larry S. Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! In *NeurIPS*,
286 2019.
- 287 [57] Abhishek Sinha, Mayank Kumar Singh, Nupur Kumari, Balaji Krishnamurthy, Harshitha
288 Machiraju, and Vineeth N. Balasubramanian. Harnessing the vulnerability of latent layers in
289 adversarially trained models. *ArXiv*, abs/1905.05186, 2019.
- 290 [58] Chawin Sitawarin, Supriyo Chakraborty, and David A. Wagner. Improving adversarial robust-
291 ness through progressive hardening. *CoRR*, abs/2003.09347, 2020.
- 292 [59] Kaustubh Sridhar, Oleg Sokolsky, Insup Lee, and James Weimer. Robust learning via persistency
293 of excitation. *ArXiv*, abs/2106.02078, 2021.
- 294 [60] Bo Sun, Nian hsuan Tsai, Fangchen Liu, Ronald Yu, and Hao Su. Adversarial defense by
295 stratified convolutional sparse coding. *2019 IEEE/CVF Conference on Computer Vision and
296 Pattern Recognition (CVPR)*, pages 11439–11448, 2019.
- 297 [61] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick
298 McDaniel. Ensemble adversarial training: Attacks and defenses. In *International Conference
299 on Learning Representations*, 2018.
- 300 [62] Jianyu Wang. Bilateral adversarial training: Towards fast training of more robust models against
301 adversarial attacks. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*,
302 pages 6628–6637, 2019.
- 303 [63] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving
304 adversarial robustness requires revisiting misclassified examples. In *International Conference
305 on Learning Representations*, 2020.
- 306 [64] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial
307 training. In *International Conference on Learning Representations*, 2020.
- 308 [65] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial
309 training. In *International Conference on Learning Representations*, 2020.
- 310 [66] Boxi Wu, Jinghui Chen, Deng Cai, Xiaofei He, and Quanquan Gu. Do wider neural networks
311 really help adversarial robustness? In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman
312 Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

- 313 [67] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust
314 generalization. In *NeurIPS*, 2020.
- 315 [68] Chang Xiao, Peilin Zhong, and Changxi Zheng. Enhancing adversarial defense by k-winners-
316 take-all. In *International Conference on Learning Representations*, 2020.
- 317 [69] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial
318 effects through randomization. In *International Conference on Learning Representations*, 2018.
- 319 [70] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in
320 deep neural networks. *ArXiv*, abs/1704.01155, 2018.
- 321 [71] Dinghui Zhang, Tianyuan Zhang, Yiping Lu, Zhanxing Zhu, and Bin Dong. *You Only*
322 *Propagate Once: Accelerating Adversarial Training via Maximal Principle*. Curran Associates
323 Inc., Red Hook, NY, USA, 2019.
- 324 [72] Haichao Zhang and Jianyu Wang. Defense against adversarial attacks using feature scattering-
325 based adversarial training. In *Advances in Neural Information Processing Systems*, 2019.
- 326 [73] Haichao Zhang and Wei Xu. Adversarial interpolation training: A simple approach for improv-
327 ing model robustness, 2020.
- 328 [74] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I.
329 Jordan. Theoretically principled trade-off between robustness and accuracy. In *International*
330 *Conference on Machine Learning*, 2019.
- 331 [75] Huan Zhang, Hongge Chen, Chaowei Xiao, Sven Gowal, Robert Stanforth, Bo Li, Duane
332 Boning, and Cho-Jui Hsieh. Towards stable and efficient training of verifiably robust neural
333 networks. In *International Conference on Learning Representations*, 2020.
- 334 [76] Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan
335 Kankanhalli. Attacks which do not kill training make adversarial learning stronger. In *ICML*,
336 2020.
- 337 [77] Jingfeng Zhang, Jianing Zhu, Gang Niu, Bo Han, Masashi Sugiyama, and Mohan Kankanhalli.
338 Geometry-aware instance-reweighted adversarial training. In *International Conference on*
339 *Learning Representations*, 2021.