Supplementary Materials: Adversarial Robustness: From Self-Supervised Pre-Training to Fine-Tuning

Tianlong Chen[1], Sijia Liu[2], Shiyu Chang[2], Yu Cheng[3], Lisa Amini[2], Zhangyang Wang[1]
[1]Texas A&M University, [2]MIT-IBM Watson AI Lab, IBM Research [3]Microsoft Dynamics 365 AI Research
{*wiwjp619,atlaswang*}*@tamu.edu,* {*sijia.liu,shiyu.chang,lisa.amini*}*@ibm.com, yu.cheng@microsoft.com*
https://github.com/TAMU-VITA/Adv-SS-Pretraining

1. Details on Self-Supervision

Here we introduce the details of the self supervision tasks (*Selfie*, *Rotation* and *Jigsaw*) used in our paper.

Selfie [8]: By masking out selected patches in an image, *Selfie* constructs a classification problem to determine the correct patch to be filled in the masked location. The training loss ℓ_p used in *Selfie* is then specified as:

$$\ell_p(\boldsymbol{\theta};\mathcal{D}_p) := \mathbb{E}_{\mathbf{x}\in\mathcal{D}_\mathrm{p}}\left[\textstyle\sum_{i\in\mathcal{I}} \ell_{\mathrm{CE}}(\boldsymbol{\theta}; S_{k\times k}(\boldsymbol{x}), i|\mathcal{A}\setminus\mathcal{I})\right], \tag{1}$$

where $S_{k\times k}(\boldsymbol{x})$ is a patch division which turns input image into $k\times k$ patches; $\ell_{\mathrm{CE}}(S_{k\times k}(\boldsymbol{x}), i|\mathcal{A}\setminus\mathcal{I};\boldsymbol{\theta})$ is the cross-entropy between the patch position output and the ground-truth label $\mathcal{I}=\{i_1,i_2,\cdots,i_k\}$, which is the index set of randomly picked "Fill in Patches"; $\mathcal{A}$ is the index set of all patches; $\mathcal{A}\setminus\mathcal{I}$ is the index difference set of "Unmasked Patches" which summarizes the content before predicting masked ones, as shown in [8]. In our case, $k=4$ for images with size 32×32; $k=7$ for images with size 224×224.

Rotation [2]: By rotating an image randomly by multiple 90 degrees, *Rotation* constructs a classification problem to determine the degree of rotation applied to an input image. The training loss ℓ_p used in *Rotation* is then specified as:

$$\ell_p(\boldsymbol{\theta};\mathcal{D}_p) := \mathbb{E}_{\mathbf{x}\in\mathcal{D}_\mathrm{p}}\left[\ell_{\mathrm{CE}}(R_r(\boldsymbol{x}), r\in\mathcal{G};\boldsymbol{\theta})\right], \tag{2}$$

where $R_r(\boldsymbol{x})$ is a rotation transformation; $\ell_{\mathrm{CE}}(R_r(\boldsymbol{x}), r\in\mathcal{G};\boldsymbol{\theta})$ is the cross-entropy between the rotation output and the ground-truth label r randomly chosen from $\mathcal{G}=\{0^\circ,90^\circ,180^\circ,270^\circ\}$.

Jigsaw [6, 1]: By dividing an image into different patches, *Jigsaw* trains a classifier to predict the correct permutation of these patches. The training loss ℓ_p used in *Jigsaw* is then specified as:

$$\ell_p(\boldsymbol{\theta};\mathcal{D}_p) := \mathbb{E}_{\mathbf{x}\in\mathcal{D}_\mathrm{p}}\left[\ell_{\mathrm{CE}}(J_{k\times k}(\boldsymbol{x}), p_j\in\mathcal{P};\boldsymbol{\theta})\right], \tag{3}$$

where $J_{k\times k}(\boldsymbol{x})$ is a process, using a $k\times k$ grid to decompose the input image in k^2 patches which are randomly shuffled and used to form an image with original size; $\ell_{\mathrm{CE}}(J_{k\times k}(\boldsymbol{x}), p_j\in\mathcal{P};\boldsymbol{\theta})$ is the cross-entropy between the permutation prediction and the ground-truth label $p_j\in\mathcal{P}$; $\mathcal{P}$ is the set of all patch permutations [1]. In our case, $|\mathcal{P}|=30$, and $k=4$.

2. More on Experiment Results

2.1. Implementation Details

Details of R-ImageNet-224 We choose 10 super classes containing a total of 190 ImageNet classes. Table 1 shows the distributions of super classes.

Table 1: Distributions of Classes in our restricted ImageNet datasets (**R-ImagetNet-224**). The class ranges are exclusive.

Super-class	Corresponding ImageNet Classes
"Dog"	151 to 268
"Cat"	281 to 285
"Frog"	30 to 32
"Bird"	80 to 100
"Fish"	389 to 397
"Insect"	300 to 319
"Ship"	510, 544, 628
"Truck"	555, 569, 675, 864, 867
"Airplane"	404 to 405, 895
"Automobile"	609, 627, 817

Hyperparameter Tuning For the hyperparameter λ in equation (3) in Section 3.3, we choose λ equal to 0.1 which is consistent with [7]. Then, we perform a grid search in the interval [0.05,0.15] and find that the results are insensitive to λ within this range.

2.2. More Results of 19 Unforeseen Attacks

We compare our proposed adversarial pretraining followed by adversarial fine-tuning approach (Ours) with the one-shot AT (adversarial training) that optimizes a classifica-

Table 2: The summary of the accuracy over unforeseen adversarial attackers. Our models are obtained after adversarial fine-tuning with adversarial *Rotation* pretraining. Baseline-*R* are co-optimized models with *Rotation* auxiliary task. The best results are highlighted (1^{st}) under each column of different unforeseen attackers. TA: Standard Testing Accuracy; RA: Robust Testing Accuracy.

Setting	TA	RA	Speckle Noise	Contrast	Glass Blur	Saturate	JPEG Compression	Impulse Noise	Gaussian Blur	Frost	Motion Blur
Ours	85.66	50.40	81.50	43.07	78.60	82.14	83.30	74.56	78.28	74.56	76.45
Baseline-*R*	83.69	51.14	80.65	37.74	75.40	80.81	80.98	75.46	73.42	69.67	71.42
Setting	Spatter	Brightness	Gaussian Noise	Pixelate	Snow	Shot Noise	Elastic Transform	Fog	Defocus Blur	Zoom Blur	
Ours	80.15	81.74	80.21	83.52	79.20	81.59	79.49	59.43	80.61	79.82	
Baseline-*R*	79.12	78.71	79.48	80.71	76.33	80.74	75.58	52.90	76.30	74.59	

Table 3: The summary of the accuracy over unforeseen adversarial attackers. Our models are obtained after adversarial fine-tuning with adversarial *Jigsaw* pretraining. Baseline-*J* are co-optimized models with *Jigsaw* auxiliary task. The best results are highlighted (1^{st}) under each column of different unforeseen attackers. TA: Standard Testing Accuracy; RA: Robust Testing Accuracy.

Setting	TA	RA	Speckle Noise	Contrast	Glass Blur	Saturate	Jpeg Compression	Impulse Noise	Gaussian Blur	Frost	Motion Blur
Ours	83.74	48.83	80.62	42.74	76.98	80.10	81.91	75.34	76.65	69.99	74.69
Baseline-*J*	79.93	51.61	75.19	37.12	72.14	77.60	77.62	70.54	72.22	70.62	69.91
Setting	Spatter	Brightness	Gaussian Noise	Pixelate	Snow	Shot Noise	Elastic Transform	Fog	Defocus Blur	Zoom Blur	
Ours	78.60	78.71	79.09	81.52	76.56	80.51	78.06	58.44	78.84	78.23	
Baseline-*J*	75.47	77.64	74.46	77.47	74.86	75.58	73.42	52.40	74.47	73.53	

Table 4: Evaluation Results (model picking through RA-best-criteria) of Two Different $(\mathcal{P}_i, \mathcal{F}_j)$ Scenarios: $\mathcal{P}_1$ (without pre-training), $\mathcal{P}_3$ (adversarial self-supervision pre-training), and $\mathcal{F}_4$ (full adversarial fine-tuning). The best results are highlighted (1^{st}) under each column of different self-supervised pretraining tasks.

Scenario	*Selfie* Pretraining			*Rotation* Pretraining			*Jigsaw* Pretraining		
	TA (%)	RA (%)	Epochs	TA (%)	RA (%)	Epochs	TA (%)	RA (%)	Epochs
$(\mathcal{P}_1, \mathcal{F}_4)$	83.96	48.49	48	83.96	48.49	48	83.96	48.49	48
$(\mathcal{P}_3, \mathcal{F}_4)$	85.88	51.07	47	85.22	51.49	48	84.34	50.11	49

tion task regularized by the self-supervised *Rotation* prediction task [3] (Baseline-*R* in Table 2) or the self-supervised *Jigsaw* task (Baseline-*J* in Table 3). Here we show more experiment results against 19 unforeseen attacks that are not used in AT [4].

As we can see, excluding a slight degradation on RA and one unforeseen attack ("Impulse Noise" in Table 2 and "Frost" in Table 3), our approach yields consistent robustness improvement in defending all 18 unforeseen attacks, where the improvement ranges from 0.73% to 6.53% in Table 2 and from 1.07% to 6.04% in Table 3. These observations strongly support the conclusions we proposed in Section 4.4.

2.3. Results of Picking the RA Best Models

Performances of two different $(\mathcal{P}_i, \mathcal{F}_j)$ scenarios, picked thorough RA-best-criteria, are collected in Table 4. $(\mathcal{P}_1, \mathcal{F}_4)$ donates the end-to-end adversarial training; $(\mathcal{P}_3, \mathcal{F}_4)$ contains adversarial self-supervision pre-training and adversarial fune-tuning. As shown in Table 4, we observe the significant RA improvement from self-supervision pre-training, which is consistent with conclusions in main context. Unfortunately, results from RA-best model picking obtain marginal computation saving (similar number of epochs). Figure 1 presents robust accuracy of four fine-tuned models along with training epochs, which offers a reference for RA-best model picking.

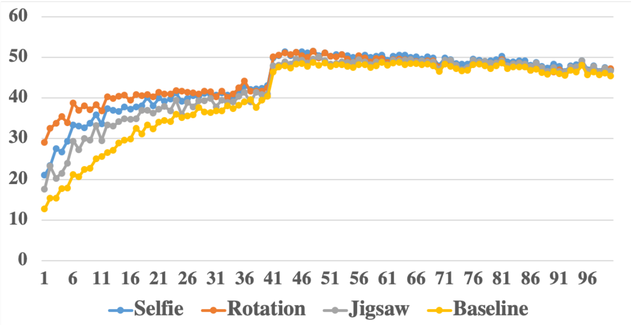

Figure 1: References for RA-best model picking. x-axis represents training **epochs** and y-axis shows the **robust accuracy** of models. Curves donate different fine-tuned models from adversarial self-supervision pre-training (*Selfie*, *Rotation*, *Jigsaw*) and random initialization (*Baseline*).

2.4. CIFAR-100 Results

In this section, we validate our proposal on CIFAR-100 datasets. $\mathcal{F}_2$ donates the partial adversarial fine-tuning and other notations are the same as those in Section 2.3. Here,

Table 5: Evaluation Results of Three Different $(\mathcal{P}_i, \mathcal{F}_j)$ Scenarios: $\mathcal{P}_1$ (without pre-training), $\mathcal{P}_2$ (standard self-supervision pretraining), $\mathcal{P}_3$ (adversarial self-supervision pre-training), $\mathcal{F}_2$ (partial adversarial fine-tuning), and $\mathcal{F}_4$ (full adversarial fine-tuning). The best results are highlighted (1^{st}) under each column of different self-supervised pretraining tasks. Here we adopt **RA-best-criteria** for model picking.

Scenario	*Rotation* Pretraining		
	TA (%)	RA (%)	Epochs
$(\mathcal{P}_1, \mathcal{F}_4)$	56.93	27.38	62
$(\mathcal{P}_3, \mathcal{F}_2)$	50.02	23.44	61
$(\mathcal{P}_3, \mathcal{F}_4)$	57.29	29.69	61

CIFAR-150K is chosen as the pre-training dataset and the subsequent fine-tuning is conducted on CIFAR-100. As shown in Table 5, observations are consistent with those on CIFAR-10, which further demonstrate the effectiveness of our approaches.

2.5. More Ensemble Results

Table 6: The vulnerability of the ensemble of fine-tuned models with *Selfie*, *Rotation* and *Jigsaw* self-supervised adversarial pretraining. The results take full adversarial fine-tuning. Each column presents ASRs under PGD attacks from different fine-tuned models.

$(\mathcal{P}_3, \mathcal{F}_4)$ Attack Evaluation	PGD attacks from Model(*Selfie*)	PGD attacks from Model(*Rotation*)	PGD attacks from Model(*Jigsaw*)
Ensemble of Model(*Selfie*) & Model(*Rotation*) & Model(*Jigsaw*)	58.25%	58.68%	58.87%

Ensemble results to different attacks As shown in Table 6, we find the ensemble of three fine-tuned models achieve similar RA on three different adversarial datasets from Model(*Selfie*), Model(*Rotation*) and Model(*Jigsaw*). It suggests that the ensemble tackles diverse vulnerability of models with different self-supervision pre-trainings, which potentially contributes an extra robustness improvement.

Table 7: Ensemble results of fine-tuned models with different standard pretrainings. $(\mathcal{P}_2, \mathcal{F}_4)$ donates the scenario of standard self-supervision and adversarial fine-tuning.

Fine-tuned Models $(\mathcal{P}_2, \mathcal{F}_4)$	TA (%)	RA (%)
Jigsaw + Rotation + Selfie	86.63	55.12

Ensemble results under $(\mathcal{P}_2, \mathcal{F}_4)$ As we can see in Table 7, another best combination, ensemble of three fine-tuned models with corresponding standard self-supervision pretrainings, yields at least 4.07% on RA while maintains a slight higher TA (+0.61%).

2.6. More Ablations

Table 8: Ablation results of the image resolution in the pretraining datasets $\mathcal{D}_{\text{p}}$. Both datasets have 30,000 images for pretraining.

Scenario	CIFAR-30K (32 × 32)			R-ImageNet-224 (224 × 224)		
	TA (%)	RA (%)	Epochs	TA (%)	RA (%)	Epochs
$(\mathcal{P}_3, \mathcal{F}_2)$	68.04	31.53	86	54.1	13.34	93
$(\mathcal{P}_3, \mathcal{F}_4)$	85.29	49.64	70	84.2	49.18	61

Ablation of the image resolution in the pretraining dataset As shown in Table 8, fine-tuned models from CIFAR-30K (32 × 32) pretraining outperform ones from R-ImageNet-224 (224 × 224), in terms of both standard and robust accuracy.

There exist two possible reasons accounting for this (somehow unexpected) observation. First, pretraining with 224 × 224 images might lead to learning features that have a scale mismatch, with the fine-tuning task on the 32 × 32 images. That is consistent with [5]'s conclusion regarding the domain (mis)match between pretraining and fine-tuning datasets. Second, a restricted set of image classes is considered in ImageNet in order to make adversarial pretraining feasible on the 224 × 224 scale. However, the restricted version of ImageNet might induce a bias during fine-tuning. We plan to conduct further experiments with both pretraining and finetuing on the 224 × 224 scale to test our conjecture.

References

[1] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019. 1

[2] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018. 1

[3] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. *arXiv preprint arXiv:1906.12340*, 2019. 2

[4] Daniel Kang, Yi Sun, Dan Hendrycks, Tom Brown, and Jacob Steinhardt. Testing robustness against unforeseen adversaries. *arXiv preprint arXiv:1908.08016*, 2019. 2

[5] Hong Liu, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Towards understanding the transferability of deep representations. *arXiv preprint arXiv:1909.12031*, 2019. 3

[6] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer, 2016. 1

[7] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. *arXiv preprint arXiv:1901.08846*, 2019. 1

[8] Trieu H Trinh, Minh-Thang Luong, and Quoc V Le. Selfie: Self-supervised pretraining for image embedding. *arXiv preprint arXiv:1906.02940*, 2019. 1