On the interplay of adversarial robustness and architecture components: patches, convolution and attention

Francesco Croce[1] **Matthias Hein**[1]

Abstract

In recent years novel architecture components for image classification have been developed, starting with attention and patches used in transformers. While prior works have analyzed the influence of some aspects of architecture components on the robustness to adversarial attacks, in particular for vision transformers, the understanding of the main factors is still limited. We compare several (non)-robust classifiers with different architectures and study their properties, including the effect of adversarial training on the interpretability of the learnt features and robustness to unseen threat models. An ablation from ResNet to ConvNeXt reveals key architectural changes leading to almost 10% higher ℓ_∞-robustness.

1. Introduction

The introduction of vision transformers (ViTs) (Dosovitskiy et al., 2021) showed that different architectures can perform on par or even better than convolutional networks (CNNs) in various computer vision tasks. This led to active research on optimizing the network design for better performance e.g. classification accuracy on the ImageNet dataset (Deng et al., 2009), which in turn resulted in several new architectures (Touvron et al., 2021a; Liu et al., 2021; Trockman & Kolter, 2022; Liu et al., 2022). It is however still partially unexplained which are the key components which make an architecture effective in a specific task (Park & Kim, 2022). It has been noticed that some architectures might natively perform better than others in side tasks, e.g. Fort et al. (2021) argue that ViTs have significantly better out-of-distribution detection ability. Moreover, recent works (Bhojanapalli et al., 2021; Paul & Chen, 2022) suggest that ViTs are more robust to common corruptions in ImageNet-C (Hendrycks & Dietterich, 2019). About robustness to adversarial attacks, Shao et al. (2021) suggest that naturally trained ViTs are more robust to ℓ_∞-bounded perturbations than ResNets. Conversely, ViTs appear more vulnerable to patch attacks (Gu et al., 2021), while the comparison is mixed on ℓ_0-attacks (Fu et al., 2022). However, Bai et al. (2021) reports that such differences might disappear with either adversarial training (Madry et al., 2018) for ℓ_∞ or using similar training protocols for the two architectures. Finally, Debenedetti (2022) has recently achieved SOTA results for the ℓ_∞-threat model on ImageNet using XCiT (El-Nouby et al., 2021), a transformer-like network which reintroduces convolutions as part of its architecture.

In this work, we first extend the analysis of the robustness of different normally trained architectures to adversarial patches: unlike previous attacks, we show that on networks which divided the input image in disjoint tokens e.g. ViTs it is preferable to position the adversarial patch to cover multiple tokens instead of a single one, although this might yield lower loss values in the case of traditional vision transformers. However, this phenomenon is largely mitigated when considering robust models. Then, we explore how the features learnt by classifiers when using adversarial training wrt ℓ_∞ differ from those of plain models, showing e.g. how the attention maps of ViTs gain in interpretability. Moreover, we study how robustness wrt ℓ_∞, generalizes to unseen attacks, both ℓ_p-bounded and not. While ResNets generally attains worse generalization, we show in an extensive ablation study regarding the transition from a ResNet to the ConvNeXt (Liu et al., 2022) architecture that small modifications in the architecture are sufficient to, at least partially, close the gap to ViTs. robust ConvNeXt which achieves 46.2% robust accuracy against l_∞-perturbations of $\epsilon = 4/255$ outperforming the recent 41.7% for the XCiT architecture Debenedetti (2022) and also being about 10% higher than a ResNet-50 architecture. However, even more interesting is that a relatively small change of the traditional ResNet-50 architecture achieves 44.0% robust accuracy.

2. Background and related works

In this section we provide the necessary background on the architectures we consider and adversarial robustness.

[1] University of Tübingen, Germany. Correspondence to: Francesco Croce <francesco.croce@uni-tuebingen.de>.

New Frontiers in Adversarial Machine Learning Workshop at 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, 2022.

2.1. Architectures

ResNets: ResNet (He et al., 2016) and WideResNet (Zagoruyko & Komodakis, 2016) rely on convolutions and on the usage of residual connections, with non-linearity provided by activation functions as ReLU and GELU (Hendrycks & Gimpel, 2016). Such models held for long time SOTA results on vision tasks, and are still dominant when considering adversarial robustness (Croce et al., 2021) for CIFAR-10 and ImageNet. ResNets do not divide the input image in disjoint patches.

Vision transformers: Dosovitskiy et al. (2021) introduced a convolution-free architecture for vision applications inspired by the transformer models used in natural language processing. ViTs split the input image in disjoint patches, in analogy of the language tokens, add a class token (`CLS`) used for classification, then process them with blocks including multi-head self-attention (Vaswani et al., 2017) and MLPs. Later, Touvron et al. (2021a) showed how it is possible to train ViTs efficiently: we use their models named DeiT with 16×16 patches.

Cross-covariance vision transformers: El-Nouby et al. (2021) replaced in the ViT design the traditional (local) self-attention with the so-called cross-covariance version of it (XCA). Moreover, each block contains two depth-wise convolutional layers to allow better communication among patches. The resulting XCiT compares favorably to DeiT with respect to performance in classification and segmentation tasks and memory usage.

ConvNeXt: To close the gap between CNNs and ViTs on ImageNet, Liu et al. (2022) modify the ResNet backbone to make it more similar to the SOTA Swin transformer (Liu et al., 2021), until they outperform it. ConvNeXt adopts patchified stem, i.e. non overlapping convolutions in the first layer, and depth-wise convolutions.

For our analysis, we use ResNet-50, WideResNet-50-2, DeiT-S XCiT-S, ConvNeXt-T, which have, except for the WideResNet, comparable size in terms of number of parameters. All are trained on the ImageNet-1k dataset and use image resolution 224×224 pixels. For naturally trained classifiers we use the checkpoints provided by either `torchvision` model zoo or the `timm` library (Wightman, 2019), while the robust ones are made available by prior works (ResNet-50 and DeiT-S are from Bai et al. (2021), WideResNet-50-2 from Salman et al. (2020), XCiT-S from Debenedetti (2022)).

2.2. Adversarial robustness

The output of a neural network can be easily modified by small perturbations of an input which do not change its semantic content (Biggio et al., 2013; Szegedy et al., 2014). Many works have focused on developing methods to find such adversarial perturbations whose size is constrained by a certain metric, e.g. an ℓ_p-norm (Carlini & Wagner, 2017), or which are limited to have a specific shape like square patches (Karmon et al., 2018) or frames (Zajac et al., 2019). At the same time, the most successful method to obtain adversarially robust models is adversarial training (Madry et al., 2018), and its recent advances (Zhang et al., 2019; Carmon et al., 2019; Gowal et al., 2020; Rebuffi et al., 2021). Gu et al. (2021); Fu et al. (2022); Lovisotto et al. (2022) have suggested that ViTs, are more vulnerable to adversarial patches than ResNets. Similarly Shao et al. (2021) argued that ViTs are more robust than other architectures in the ℓ_∞-threat model, However, Bai et al. (2021) report that, when trained with similar augmentations, DeiTs and ResNets have similar robustness to adversarial patches, and using GELU in the ResNet backbone suffices to obtain with adversarial training classifiers as robust as DeiTs with respect to ℓ_∞-perturbations with size $\epsilon = 4/255$ on ImageNet.

3. Robustness of natural and robust models to adversarial patches

In this section we analyze the interaction between patch attacks and the token grid used by DeiT and XCiT, Then, by developing a simple greedy attack, we show that both ViTs and ResNet are less robust to adversarial patches than what has been reported by previous works.

3.1. Effect of grid aligned and non-grid aligned patches on adversarial loss

Both Fu et al. (2022); Gu et al. (2021) evaluate the robustness of vision transformers using mostly patches which are aligned with the grid of the input tokens, so that one adversarial patch exactly covers one cell of the tokens grid. However, adversarial attacks might benefit from modifying multiple tokens by a smaller fraction: we consider patches which are centered at the intersection point of the tokens grid (each patch covers 1/4 of 4 contiguous tokens). For transformers with 16×16 tokens (and using patches of the same size) this yields $14 \times 14 = 196$ possible positions of *aligned* patches and $13 \times 13 = 169$ *non aligned* ones. For each one we maximize the margin loss (Carlini & Wagner, 2017) for 100 iterations with APGD (Croce & Hein, 2020). In Fig. 1 we show the results for models trained either naturally (left) or with adversarial training wrt ℓ_∞ (right). Among the naturally trained models, for DeiT-S the grid-aligned patches attain much higher loss values than the non aligned ones, while both yielding 0% robust accuracy (see below). Conversely, there is no particular difference when using ResNet-50 or XCiT-S, which also use convolutional layers. The adversarially trained DeiT-S does not show the same behavior of the plain one. Moreover, for all adversarially trained architectures, the attacks achieve higher loss at patch locations in correspondence of class-specific features in the image, creating a sort of saliency map, in strong contrast to non robust classifiers. This is in line with the observation that robust models have gradients significantly aligned with human perception (Tsipras et al., 2019; Santurkar et al., 2019).

3.2. Robustness of plain and robust classifiers to adversarial patches

A simple patch attack: several prior works (Brown et al., 2017; Yang et al., 2020; Croce et al., 2022) have introduced patch attacks. Here we adopt a more greedy strategy which allows us to compare the effectiveness of patches aligned or not with the token grid. For both aligned patches and not aligned one, we optimize the margin loss for 20 iterations with APGD (ℓ_∞-version, with initial step size 0.5) for all possible patches then we select the 20% of patches of each type with highest loss and optimize their content for 480 iterations (the location of the patches remains unchanged).

Robustness to aligned and non aligned patches: Table 1 shows the robust accuracy to aligned and not aligned patches, together with the worst-case over them. First we analyze the plain models: for ResNet-50 the aligned and non-aligned perform similarly as expected since there is no tokenization. On DeiT-S both perform equally well, despite the aligned patches achieving significantly higher loss values on average. Finally, for XCiT-S, the non aligned patches achieve higher success rate than the aligned ones. Overall, XCiT-S appears more robust than both DeiT-S and ResNet-50, implying that transformer-based architectures are not necessarily more vulnerable to adversarial patches. Moreover, our greedy attack yields much lower robust accuracy for both DeiT-S and ResNet-50 compared to what has been reported by Fu et al. (2022), that is 6.25% and 24.00% respectively (although different subsets of test points are used). For adversarially trained classifiers wrt the ℓ_∞-threat model, the robust accuracy for all architectures against patch attacks improves, with DeiT-S being the most robust one, and there is no clear trend regarding the comparison of aligned and non-aligned patches.

Table 1. Robust accuracy (%) of different ImageNet models (computed for 1000 images) to patch perturbations which are either aligned or non-aligned with the tokens grid (for ResNet-50 the grid of the transformers is used). Worst-case is over both patches.

patch type	*plain training*			*adversarial training*		
	DeiT-S	XCiT-S	RN-50	DeiT-S	XCiT-S	RN-50
aligned	0.0	6.7	1.7	24.7	20.4	15.3
not align.	0.0	4.9	1.8	22.5	20.1	17.4
worst-case	0.0	4.1	1.2	21.4	18.7	14.5

Robust to patches at a fixed-position: Lovisotto et al. (2022) have recently reported that DeiT-B, the larger version of DeiT-S, has a non trivial robustness of 13.1% when using a 16×16 patch at a fixed position, i.e. in the top left corner. However, we managed, by using our attack which optimizes the margin loss with APGD, to reduce it to 0%.

4. Effect of adversarial training on ViTs features

We now study how the interpretability of the representations learnt by vision transformers is affected by adversarial training, with ℓ_∞-threat model with radius $\epsilon = 4/255$.

4.1. Interpretability of attention maps

In Fig. 2 we show for the DeiT-S classifiers the attention maps of the CLS token for each head in the last block: in each row, the first 6 maps are produced with the plain model, while the remaining 6 are from the robust one, all corresponding to the original image shown on the far left. The analogous maps for XCiT are in Fig. 5 in appendix. In both cases, the maps of the robust models are significantly more interpretable than those of the plain ones: the various heads are triggered by different parts (and objects) of the image, unlike those from the plain classifier which are mostly sparse and similar to each other. Interestingly, this is similar to what happens to the attention maps of models trained by self-supervision with DINO (Caron et al., 2021).

4.2. Inner representations of XCiT

El-Nouby et al. (2021) noticed that the norm across the features dimension of queries and keys of each token may indicate the salient regions in the image (higher values correspond to more salient areas). We show this for the plain model in the top part of Fig. 3 for the keys of the last XCA block (each column corresponds to a head), using images of resolution 384×384. While these identify meaningful features, they also show highly activated patches at random positions. This effect is instead absent for the robust model (bottom part of Fig. 3), whose images appear “denoised”, with different heads focusing on complementary details of the image. We plot the maps of the robust model for three images containing dogs in Fig. 4: the heads focus on similar features across images (see also the appendix).

5. Generalization of robustness to unseen threat models

It is known that robustness in a specific threat model does not necessarily generalizes to other ones, e.g. across ℓ_p-norms (Tramèr & Boneh, 2019; Kang et al., 2019). We study here to which extent this varies across architectures, and test how the classifiers with adversarial training wrt ℓ_∞ behave

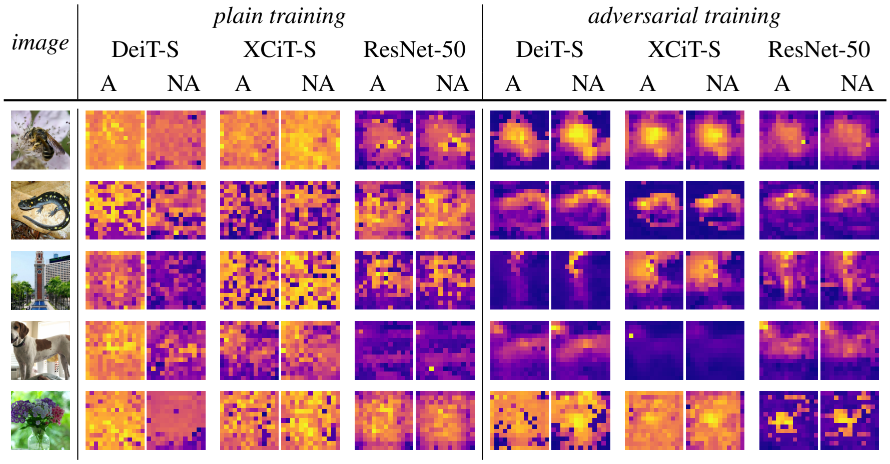


Figure 1. For each image and model we show the map of the values obtained maximizing the margin loss within disjoint patches of size 16 × 16 pixels which are either aligned (A) with the grid of the tokens of ViTs or not (NA) i.e. centered on the intersection the grid (that is having a 4 × 4 overlap with the four adjacent patches). Each pair (aligned and not) of plots is normalized independently. Brighter colors indicate higher values of the loss.

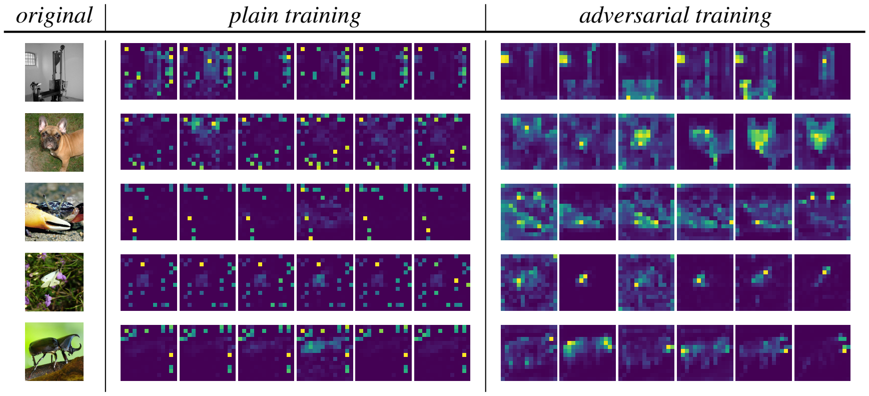
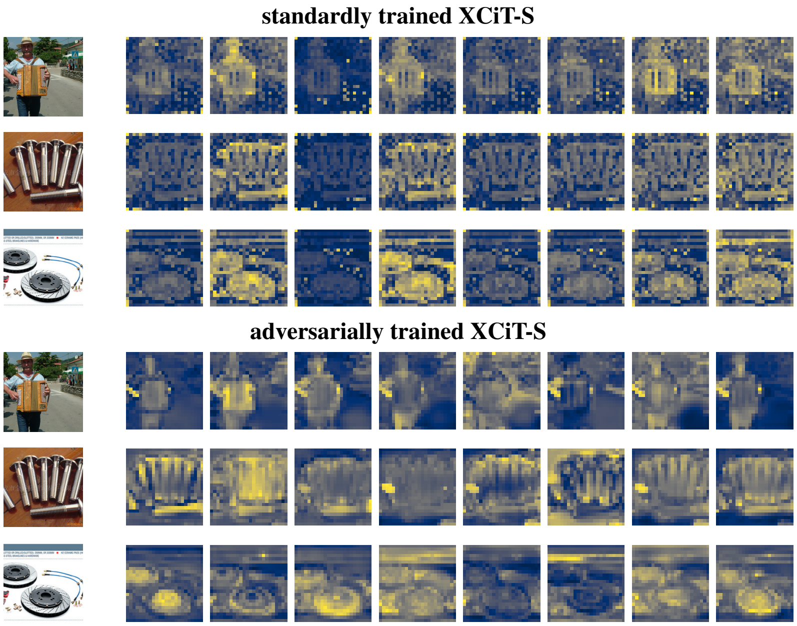


Figure 2. Attention maps of the `CLS` token for each head of the last layer of a DeiT-S model The attention of the robust DeiT-S is concentrated on the object, where each head pays attention to different parts.

in other, unseen, threat models. This might hint to which architectural components are relevant for designing more robust models. In fact, the results reported in Sec. 3 already suggest that not all network trained to be robust wrt ℓ_∞ are equally vulnerable to adversarial patches. Moreover, this allows to check whether some architectures are preferable wrt the robustness in the threat model used for training i.e. ℓ_∞.

Experimental setup: We report all statistics on 1000 points. For the ℓ_∞, ℓ_2 and ℓ_1 threat models we use bounds 4/255, 0.5 and 75 respectively, and APGD for cross-entropy and DLR-loss from AutoAttack (Croce & Hein, 2020) as attacks. We test robustness wrt ℓ_0 in pixel space (that is all color channels of a selected pixels are perturbed) for bound $\epsilon_0 = 100$ using `Sparse-RS` (Croce et al., 2022) with 50,000 queries, after running the white-box but less effective PGD$_0$ (Croce & Hein, 2019) to quickly reduce the points to test. For adversarial patches we keep the same setup as used in Sec. 3, i.e. 16 × 16 perturbations, with our greedy attack. Finally, we use adversarial frames of width 2 pixels: in this case we run 100 iterations and 5 restarts of APGD on the margin loss. In addition to the models introduced in Sec. 2.1, we train, with 100 epochs of adversarial training wrt ℓ_∞ and $\epsilon = 4/255$ (details in the appendix), a ConvNeXt-T and a modification of ResNet-50 which includes patchified stem and depth-wise convolutions (see Sec. 6 for details). Further, we retrain a ResNet-50 with GELU, as in Bai et al. (2021), in the same setup as the ConvNeXt-T for further comparison.

Results: Table 2 shows that our trained ConvNeXt has 4.5% higher robust accuracy in ℓ_∞ than the XCiT-S (for which significant optimization of hyperparameters has been performed in Debenedetti (2022)) and 9.7% more robust than the ResNet-50. Remarkably, "ResNet-50 modified", where


standardly trained XCiT-S

adversarially trained XCiT-S

Figure 3. Visualization of the norm across the features dimension of the keys of the last XCA block, with each column representing a head, for a normally trained model (top) and an adversarially trained model (bottom). Images are with resolution 384×384.

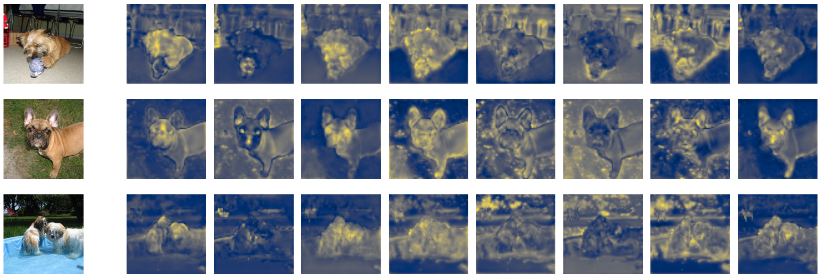


Figure 4. Comparison of the visualization of norm across the features dimension of the keys of the last XCA block of the robust XCiT-S for different images of resolution 1024×1024 of similar classes: similar features of the dog are highlighted across images.

patchified stem and depth-wise convolutions have been introduced, performs almost as good as the ConvNeXt in ℓ_∞. When looking at the generalization to unseen threat models, DeiT-S and XCiT-S attain the best results, especially wrt ℓ_p-norms. Moreover, "ResNet-50 modified" improves most of the time the performance over ResNets and ConvNeXt, but is worse than the transformer-based models. It is an interesting open question if one can combine the advantages of transformers and improved ResNet-architecture further to define an even better architecture for adversarial robustness.

6. From ResNet to ConvNeXt for robustness

Following the observations of Sec. 5 we explore which components of the transformers-based architectures might lead i) to better robustness into the threat model used for training (ℓ_∞) and ii) better generalization of robustness to unseen threat model, which is relatively weak for ResNets. We follow the steps of Liu et al. (2022) and start from the basic version of ResNet-50 (as implemented in `torchvision`) and progressively update it. For each resulting architecture we train a plain model and use it as initialization for single step adversarial training with $\epsilon = 4/255$ in the ℓ_∞-threat model (this improves the clean accuracy of robust models with the short training). Given the high computational cost, we use the `FFCV` library (Leclerc et al., 2022) for preprocessing the dataset, train for 16 epochs with batch size of 2048 (reduced only when necessary to fit a model into GPU memory), cyclic schedule for the learning rate with maximum value of 0.004, AdamW (Loshchilov & Hutter, 2019) as optimizer, weight decay of 0.05 (more details in appendix). Table 3 tracks the clean accuracy and robust accuracy wrt ℓ_∞ and ℓ_2 (as a proxy for generalization to other threat models), with bounds of 4/255 and 0.5 respectively,

Table 2. Robust accuracy on 1000 points of ℓ_∞-robust models to ℓ_p-bounded and sparse attacks.

model	*reference*	*seen*		*unseen*				
		clean	ℓ_∞	ℓ_2	ℓ_1	ℓ_0	*patches*	*frames*
ResNet-50	Bai et al. (2021)	68.2	36.7	15.6	3.1	5.7	14.5	19.7
WideResNet-50-2	Salman et al. (2020)	69.2	38.2	20.2	4.1	5.1	16.4	19.7
DeiT-S	Bai et al. (2021)	66.4	35.6	40.1	21.7	24.5	21.4	25.8
XCiT-S	Debenedetti (2022)	72.8	41.7	45.3	22.2	20.8	18.7	21.6
ResNet-50	retrained	68.9	36.7	19.2	3.9	6.6	16.3	24.3
ResNet-50 modified	retrained	69.9	44.0	34.3	11.2	14.5	17.1	20.8
ConvNeXt-T	retrained	70.7	46.2	30.6	9.2	16.4	21.9	18.7

Table 3. Robust accuracy on 1000 points to ℓ_p-bounded perturbations of models with different architectures adversarially trained wrt ℓ_∞.

model	*clean*	ℓ_∞	ℓ_2
ResNet-50	60.7	28.1	16.7
3:3:9:3 stage ratio	62.0	27.5	18.5
ReLU → GELU	61.8	29.6	13.6
depth-wise conv. with increased width	63.0	28.5	19.6
patchify stem	61.4	27.4	35.4
patchify stem + depth-wise conv. with increased width	63.4	29.1	36.9
patchify stem + GELU	64.4	33.4	37.6
patchify stem + GELU + depth-wise conv. with increased width	64.6	35.0	38.2
ResNet-50 + patchify stem + GELU + depth-wise conv. with increased width	64.6	35.0	38.2
+ 3:3:9:3 stage ratio	66.3	35.5	39.3
+ inverted bottleneck	66.0	33.3	28.9
+ fewer activations and normalizations	64.6	31.5	37.0
+ BatchNorm → LayerNorm	62.6	34.0	39.2
+ move downsampling to a separate layer	64.1	34.4	37.7
ConvNeXt-T without Layer Scale	65.2	36.5	36.7
ConvNeXt-T	65.2	37.9	29.5

of each model, evaluated with APGD for cross-entropy and DLR loss from AutoAttack.

Effect of main architecture components: We start with testing individually the effect of the main modifications brought by Liu et al. (2022): 1) changing the number of residual blocks in each stage, 2) using GELU as activation function instead of ReLU, 3) using depth-wise convolutions together with increasing of $1.5\times$ the width to roughly preserve the number of parameters, 4) patchified stem. Note that Liu et al. (2022) modify the activation function only later on in their road towards ConvNeXt, and it has a limited influence on the clean accuracy. However, Xie et al. (2020); Bai et al. (2021) have shown that a smooth activation function might have a significant impact on robustness, therefore we include it among the initial main components. Table 3 confirms that using GELU improves ℓ_∞-robustness, although it decreases it for ℓ_2. Conversely, the patchified stem notably improves the robustness wrt ℓ_2 compared to the baseline without modifying that for ℓ_∞. Combining the patchified stem and GELU leads to improvements in all statistics, with +5% of robust accuracy wrt ℓ_∞ and almost +20% in that wrt ℓ_2. This shows that two simple modifications of the architecture can largely influence the effectiveness of adversarial training. A small improvement in all metrics is achieved further adding the depth-wise convolutions: this yield the "ResNet-50 modified" used in Sec. 5. We conjecture that the patchified stem improves robustness wrt ℓ_2 because it implicitly performs a sort of dimensionality reduction operating on disjoint subsets of input dimensions. Then optimizing wrt adversarial ℓ_∞-perturbations should lead effectively to parameters which work well wrt ℓ_2 too, since due to the dimensionality reduction the threat models become more comparable (note that it holds $\|x\|_\infty \leq \|x\|_2 \leq \sqrt{d}\,\|x\|_\infty$ where d is the dimension).

From improved ResNet to ConvNeXt: Starting from this modified version of ResNet-50, we progressively take several further steps to reach the ConvNeXt definition. In the second part of Table 3 one observes that the remaining modifications have smaller effect on all statistics, although cumulatively bring additional improvements especially for ℓ_∞-robustness. Moreover, we notice that Layer Scale (Touvron et al., 2021b) has a significant effect on the ConvNeXt, improving ℓ_∞-robustness at cost of lower robust accuracy wrt ℓ_2. Overall, this process shows that ConvNeXt is a better suited architecture for adversarial robustness than the original ResNet for ℓ_∞-robustness, but might be improved for generalization to unseen threat models.

References

Bai, Y., Mei, J., Yuille, A., and Xie, C. Are transformers more robust than CNNs? In *NeurIPS*, 2021.

Bhojanapalli, S., Chakrabarti, A., Glasner, D., Li, D., Unterthiner, T., and Veit, A. Understanding robustness of transformers for image classification. In *CVPR*, 2021.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. In *ECML/PKKD*, 2013.

Brown, T. B., Mané, D., Roy, A., Abadi, M., and Gilmer, J. Adversarial patch. In *NeurIPS 2017 Workshop on Machine Learning and Computer Security*, 2017.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, 2017.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. S. Unlabeled data improves adversarial robustness. In *NeurIPS*, pp. 11190–11201. 2019.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

Croce, F. and Hein, M. Sparse and imperceivable adversarial attacks. In *ICCV*, 2019.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.

Croce, F., Andriushchenko, M., Sehwag, V., Flammarion, N., Chiang, M., Mittal, P., and Hein, M. Robustbench: a standardized adversarial robustness benchmark. In *NeurIPS Datasets and Benchmarks Track*, 2021.

Croce, F., Andriushchenko, M., Singh, N. D., Flammarion, N., and Hein, M. Sparse-rs: a versatile framework for query-efficient sparse black-box adversarial attacks. In *AAAI*, 2022.

Debenedetti, E. Adversarially robust vision transformers. Master's thesis, Swiss Federal Institue of Technology, Lausanne (EPFL), 4 2022.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

El-Nouby, A., Touvron, H., Caron, M., Bojanowski, P., Douze, M., Joulin, A., Laptev, I., Neverova, N., Synnaeve, G., Verbeek, J., et al. Xcit: Cross-covariance image transformers. In *NeurIPS*, 2021.

Fort, S., Ren, J., and Lakshminarayanan, B. Exploring the limits of out-of-distribution detection. In *NeurIPS*, 2021.

Fu, Y., Zhang, S., Wu, S., Wan, C., and Lin, Y. Patch-fool: Are vision transformers always robust against adversarial perturbations? In *ICLR*, 2022.

Gowal, S., Qin, C., Uesato, J., Mann, T., and Kohli, P. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593v2*, 2020.

Gu, J., Tresp, V., and Qin, Y. Are vision transformers robust to patch perturbations? *arXiv preprint arXiv:2111.10659*, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. In *ECCV*, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

Hendrycks, D. and Gimpel, K. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Kang, D., Sun, Y., Brown, T., Hendrycks, D., and Steinhardt, J. Transfer of adversarial robustness between perturbation types. *arXiv preprint, arXiv:1905.01034*, 2019.

Karmon, D., Zoran, D., and Goldberg, Y. Lavan: Localized and visible adversarial noise. In *ICML*, 2018.

Leclerc, G., Ilyas, A., Engstrom, L., Park, S. M., Salman, H., and Madry, A. ffcv. `https://github.com/libffcv/ffcv/`, 2022. commit xxxxxxx.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. A convnet for the 2020s. *CVPR*, 2022.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *ICLR*, 2019.

Lovisotto, G., Finnie, N., Munoz, M., Mummadi, C. K., and Metzen, J. H. Give me your attention: Dot-product attention considered harmful for adversarial patch robustness. In *CVPR*, 2022.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Park, N. and Kim, S. How do vision transformers work? In *ICLR*, 2022.

Paul, S. and Chen, P.-Y. Vision transformers are robust learners. In *AAAI*, 2022.

Rebuffi, S.-A., Gowal, S., Calian, D. A., Stimberg, F., Wiles, O., and Mann, T. Data augmentation can improve robustness. In *NeurIPS*, 2021.

Salman, H., Ilyas, A., Engstrom, L., Kapoor, A., and Madry, A. Do adversarially robust imagenet models transfer better? In *ArXiv preprint arXiv:2007.08489*, 2020.

Santurkar, S., Tsipras, D., Tran, B., Ilyas, A., Engstrom, L., and Madry, A. Image synthesis with a single (robust) classifier. In *NeurIPS*, 2019.

Shao, R., Shi, Z., Yi, J., Chen, P.-Y., and Hsieh, C.-J. On the adversarial robustness of vision transformers. *arXiv preprint arXiv:2103.15670*, 2021.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *ICLR*, pp. 2503–2511, 2014.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021a.

Touvron, H., Cord, M., Sablayrolles, A., Synnaeve, G., and Jégou, H. Going deeper with image transformers. In *ICCV*, 2021b.

Tramèr, F. and Boneh, D. Adversarial training and robustness for multiple perturbations. In *NeurIPS*, 2019.

Trockman, A. and Kolter, J. Z. Patches are all you need?, 2022. URL https://openreview.net/forum?id=TVHS5Y4dNvM.

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. In *ICLR*, 2019.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

Wightman, R. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Wong, E., Rice, L., and Kolter, J. Z. Fast is better than free: Revisiting adversarial training. In *ICLR*, 2020.

Xie, C., Tan, M., Gong, B., Yuille, A., and Le, Q. V. Smooth adversarial training. *arXiv preprint arXiv:2006.14536*, 2020.

Yang, C., Kortylewski, A., Xie, C., Cao, Y., and Yuille, A. Patchattack: A black-box texture-based attack with reinforcement learning. In *ECCV*, 2020.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In *BMVC*, 2016.

Zajac, M., Zołna, K., Rostamzadeh, N., and Pinheiro, P. O. Adversarial framing for image and video classification. In *AAAI*, 2019.

Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. Theoretically principled trade-off between robustness and accuracy. In *ICML*, 2019.

A. Inner representations of adversarially trained ViTs

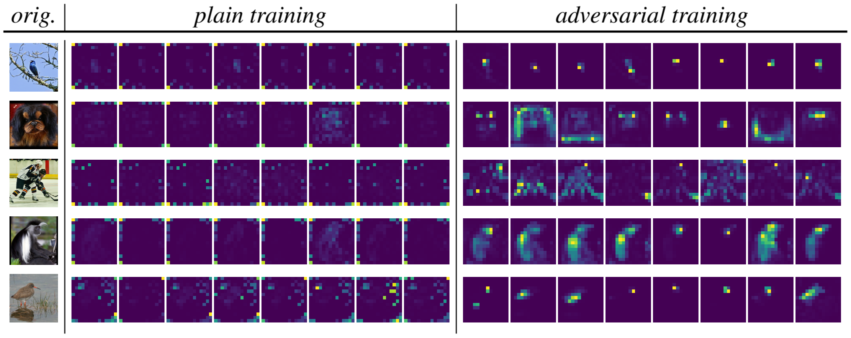


Figure 5. Attention maps of the `CLS` token for each head of the last layer of a XCiT-S model: in the first column we show the original image, then the 8 maps for the normally trained model, finally the same for the adversarially trained one. Similar to the DeiT-S in Figure 2, the attention maps of the robust XCiT-S model are concentrated on the object and highlight different parts of the object.

As shown in Sec. 4.2, it is possible to visualize salient regions of an image with the keys matrices of the XCA blocks of the XCiT models, with adversarially trained models yielding more interpretable maps. We provide further examples of this in Fig. 6 with images in resolution 1024×1024 with similar contents. Moreover, we generate similar maps using the queries matrices (instead of keys): Fig. 7 shows that even in this case for each head only salient regions are triggered.

B. Experimental details

In the following we provide additional details about the setup of the various experiments and further results.

B.1. Generalization of robustness to unseen threat models

Experimental setup: For the additional classifiers we perform adversarial training wrt ℓ_∞ at $\epsilon = 4/255$ adapting the pipeline of `https://github.com/facebookresearch/deit`. We train for 100 epochs with batch size of 1024, AdamW optimizer, cosine learning rate with maximum value of 0.001 after 10 epochs of warm-up (during which the learning rate is lineraly increased) and 5 of cool-down, weight decay of 0.05. As noted by Debenedetti (2022), for adversarial training it is not necessary to use heavy augmentation techniques. Since both the "ResNet-50 modified" and ConvNeXt-T suffer from catastrophic overfitting (Wong et al., 2020) when using FGSM without random initialization, we prevent it by increasing the number of steps for the inner maximization in adversarial training (up to 3 for former, up to 2 for the latter). Therefore we retrain the standard ResNet-50 with GELU, as in Bai et al. (2021), with 2 steps to match the budget given to ConvNeXt-T. We note that its results are consistent with the ResNet-50 from Bai et al. (2021) trained with single step adversarial training (see Table 2). For all architectures we use a plain model, trained for the ablation study of ResNet-50 (see Sec. 6), as initialization. We select the model among the checkpoints at different epochs as the most robust one to FGSM attack on 5000 images.

Additional results: We additionally test the robustness of the various models reported in Table 2 in the ℓ_p-threat models with the full version of AutoAttack (Croce & Hein, 2020) and report the results in Table 4, where one can see that the robust accuracy decreases at most by 0.3% by using the missing attacks in AutoAttack, this is FAB- and Square-Attack.

Table 4. Robust accuracy on 1000 points of ℓ_∞-robust models to ℓ_p-bounded with AutoAttack.

model	*reference*	*seen*		*unseen*	
		clean	ℓ_∞	ℓ_2	ℓ_1
ResNet-50	Bai et al. (2021)	68.2	36.7	15.4	3.1
WideResNet-50-2	Salman et al. (2020)	69.2	38.2	19.9	4.1
DeiT-S	Bai et al. (2021)	66.4	35.6	40.1	21.5
XCiT-S	Debenedetti (2022)	72.8	41.7	45.0	22.0
ResNet-50	retrained	68.9	36.7	18.9	3.9
ResNet-50 modified	retrained	69.9	44.0	34.2	11.2
ConvNeXt-T	retrained	70.7	46.2	30.6	9.2

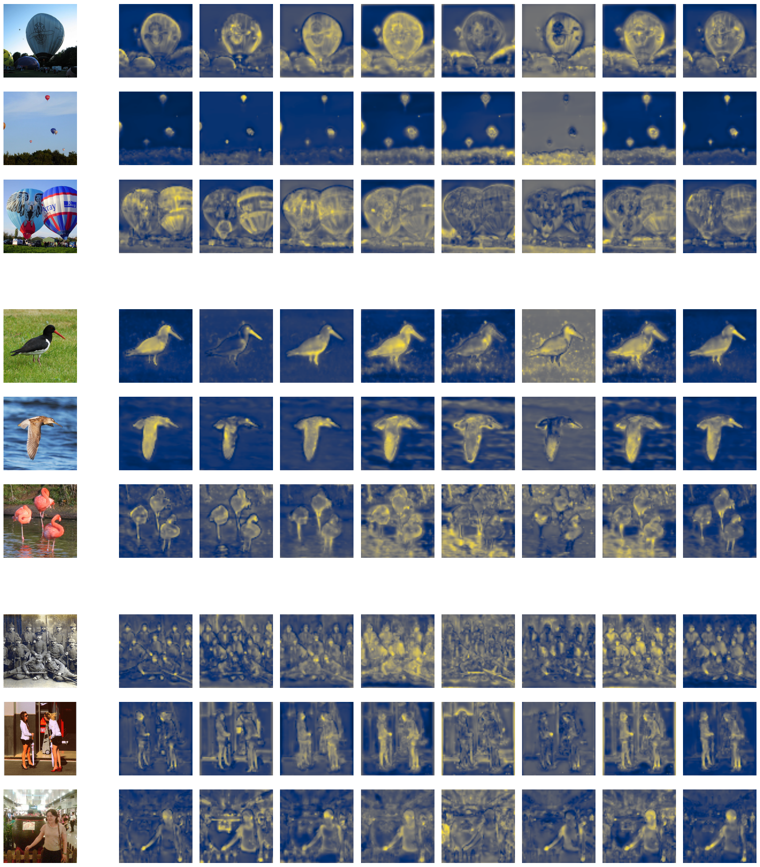


Figure 6. Visualization of norm across the features dimension of the keys of the last XCA block of the robust XCiT-S for different images of resolution 1024×1024 of similar classes.

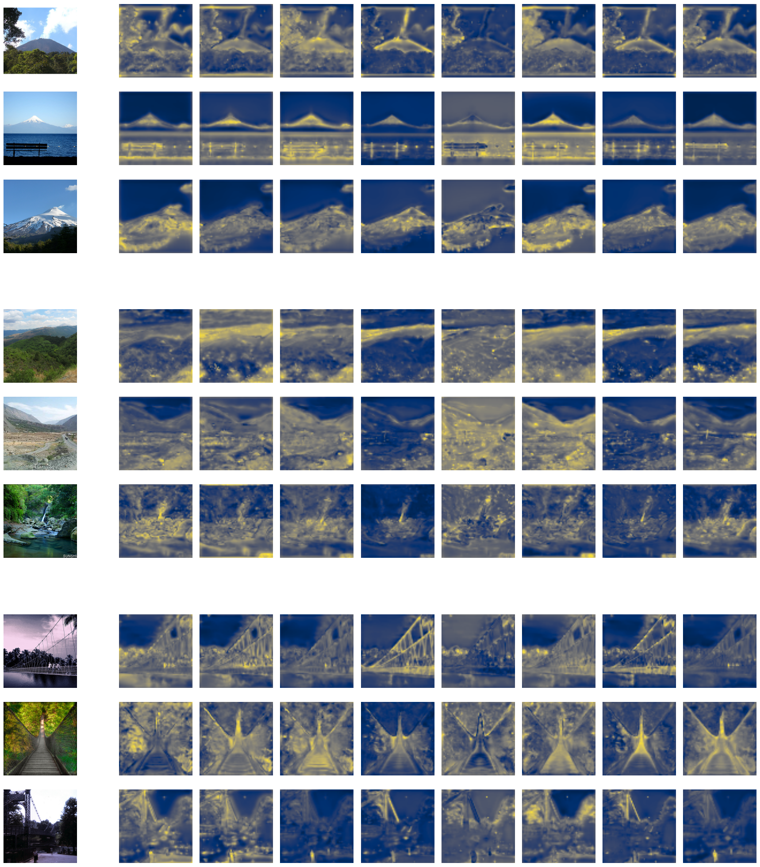


Figure 7. Visualization of norm across the features dimension of the queries of the last XCA block of the robust XCiT-S for different images of resolution 1024×1024 of similar classes.

B.2. From ResNet to ConvNeXt for robustness

Experimental setup: For the models reported in Sec. 6 we rely on the FFCV library (Leclerc et al., 2022) since it provides a significant speed-up of training. In particular, we follow the training script, adapted to the adversarial training setup, available at https://github.com/libffcv/ffcv-imagenet. In particular, we use the standard pre-processing with the exception of the maximum size of the images which we set to 400 pixels instead of 500 (for the short training with only 16 epochs this does not appear to degrade performance). Moreover, we disable BlurPool and test-time augmentation. For the inner maximization process in adversarial training we use FGSM without random initialization.

Additional results: We report in Table 5 the clean accuracy (on the 1000 points used for the evaluation of robustness) of the classifiers naturally trained with the various architectures and used as initialization for the robust models shown in Table 3. We observe that all models have clean accuracy above 70% and in the same range. Note the short training scheme we employ here is very different from the setup of Liu et al. (2022) where ConvNeXt significantly outperforms ResNet.

Table 5. We report the clean accuracy (on 1000 points) of the models with different architectures obtained with plain training.

model	*clean*
ResNet-50	70.9
3:3:9:3 stage ratio	71.1
ReLU → GELU	72.6
depth-wise conv. with increased width	73.2
patchify stem	72.0
patchify stem + depth-wise conv. with increased width	73.9
patchify stem + GELU	72.7
ResNet-50 + patchify stem + GELU + depth-wise conv. with increased width	72.5
+ 3:3:9:3 stage ratio	73.9
+ inverted bottleneck	71.0
+ fewer activations and normalizations	71.2
+ BatchNorm → LayerNorm	71.8
+ move downsampling to separate layer	72.5
ConvNeXt-T without Layer Scale	72.0
ConvNeXt-T	71.8

C. Visualization of adversarial perturbations

Bhojanapalli et al. (2021) noticed that adversarial perturbations generated for ViTs show a grid structure which resembles that of the tokens. We show in Fig. 8 the adversarial perturbations (summed over color channels) generated for the ℓ_2-threat model ($\epsilon = 2$ is used) for each naturally trained classifier. The grid structure appears for the models using input tokenization. However, among those, the effect looks stronger for DeiT, which does not use any convolutional component, and milder for XCiT, which relies on both convolutional layers and cross-covariance self-attention.

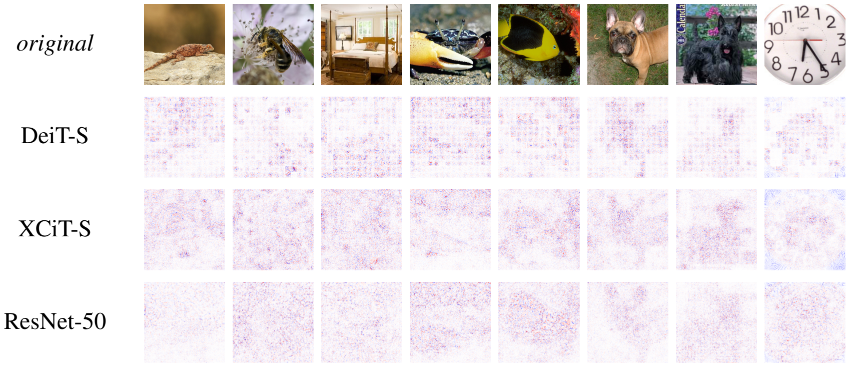


Figure 8. We plot the adversarial perturbations wrt ℓ_2 of different natural models: the perturbations are summed over color channels and rescaled so that white areas correspond to zero entries, blue negative, red positive (with intensity indicating their magnitude).