ARE VISION TRANSFORMERS ROBUST TO PATCH-WISE PERTURBATION?

Anonymous authors
Paper under double-blind review

ABSTRACT

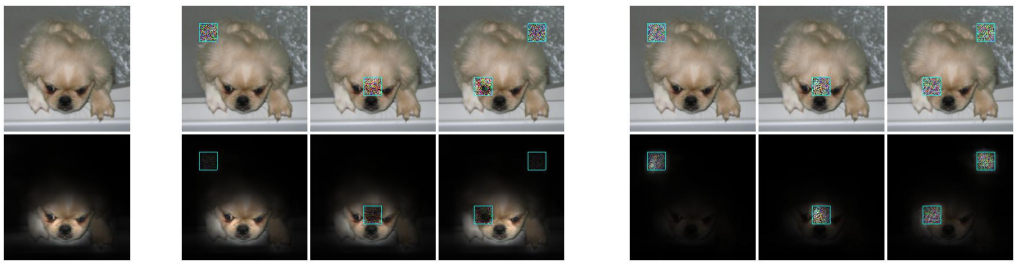
The recent advances in Vision Transformer (ViT) have demonstrated its impressive performance in image classification, which makes it a promising alternative to Convolutional Neural Network (CNN). Unlike CNNs, ViT represents an input image as a sequence of image patches. The patch-wise input image representation makes the following question interesting: How does ViT perform when individual input image patches are perturbed with natural corruptions or adversarial perturbations, compared to CNNs? In this work, we conduct a comprehensive study on the robustness of vision transformers to patch-wise perturbations. Surprisingly, we find that vision transformers are more robust to naturally corrupted patches than CNNs, whereas they are more vulnerable to adversarial patches. Based on extensive qualitative and quantitative experiments, we discover that ViT's stronger robustness to natural corrupted patches and higher vulnerability against adversarial patches are both caused by the attention mechanism. Specifically, the attention model can help improve the robustness of vision transformers by effectively ignoring natural corrupted patches. However, when vision transformers are attacked by an adversary, the attention mechanism can be easily fooled to focus more on the adversarially perturbed patches and cause a mistake.

1 INTRODUCTION

Recently, Vision Transformer (ViT) has achieved impressive performance (Dosovitskiy et al., 2020; Touvron et al., 2021), which makes it become a potential alternative to convolutional neural networks (CNNs). Unlike CNNs, ViT processes the input image as a sequence of image patches. Then, a self-attention mechanism is applied to aggregate information from all patches. Existing works have shown that ViT are more robust than the popular CNNs when the whole input image is perturbed with natural corruptions or adversarial perturbations (Bhojanapalli et al., 2021). In this work, instead, we study the robustness of ViT to patch-wise perturbations based on its special patch-based architecture.

Two typical types of perturbations are considered to compare the robustness between ViTs and CNN (e.g., ResNets (He et al., 2016)). One is natural corruptions (Hendrycks & Dietterich, 2019), which is to test models' robustness under distributional shift. The other is adversarial perturbations (Szegedy et al., 2014), which are created by an adversary to specifically fool a model to make a wrong prediction. Surprisingly, we find ViT does not always perform more robustly than ResNet. When individual image patches are naturally corrupted, ViT performs more robust than ResNet. However, when input image patch(s) are adversarially attacked, ViT shows a higher vulnerability.

Digging down further, we find the reason behind is that the self-attention mechanism of ViT can effectively ignore the natural patch corruption, while it's also easy to manipulate the self-attention mechanism to focus on an adversarial patch. This is well supported by rollout attention visualization (Abnar & Zuidema, 2020) on ViT. As shown in Fig. 1 (a), ViT successfully attends to the class-relevant features on the clean image, i.e., the head of the dog. When one or more patches are perturbed with natural corruptions, shown in Fig. 1 (b), ViT can effectively ignore the corrupted patches and still focus on the main foreground to make a correct prediction. In Fig. 1 (b), the attention weights on the positions of naturally corrupted patches are much smaller even when the patches appear on the foreground. In contrast, when the patches are perturbed with adversarial perturbations by an adversary, shown in Fig. 1 (c), ViT is successfully fooled to make a wrong prediction because the attention of ViT is misled to focus on the adversarial patches instead.



(a) Clean Image (b) with Naturally Corrupted Patch (c) with Adversarial Patch

Figure 1: Images with patch-wise perturbations (top) and their corresponding attention maps (bottom). The attention mechanism in ViT can effectively ignore the naturally corrupted patches to maintain a correct prediction, whereas it is forced to focus on the adversarial patches to make a mistake. The images with corrupted patches are all correctly classified. The images with adversary patches in subfigure 1c are misclassified as *dragonfly*, *axolotl*, and *lampshade*, respectively.

Based on the patch-based architectural structure of vision transformers, we further investigate the sensitivity of ViT against patch positions and patch alignment of adversarial patches. First, we discover that ViT is insensitive to different patch positions, while ResNet shows high vulnerability on the central area of input images and much less on corners. We attribute this to the architecture bias of ResNet where pixels in the center can affect more neurons than the ones in corners. In contrast, each patch within ViT can equally interact with other patches regardless of its position. Further, we find that for ViT, the adversarial patch designed to attack one particular position can successfully transfer to other positions of the same image as long as they are aligned with input patches. In contrast, the ones on ResNet hardly do.

Our main contributions can be summarized as follows:

- We discover that ViT is more robust to natural patch corruption than ResNet, whereas it is more vulnerable to adversarial patch perturbation.
- Based on extensive analysis, we find that the self-attention mechanism, the core building block of vision transformers, can effectively ignore natural corrupted patches to maintain a correct prediction but be easily fooled to focus on adversarial patches to make a mistake.
- We show that ViT and ResNet exhibit different sensitivities against patch positions and patch alignment of adversarial patch attacks due to their different architectural structures.

2 Related Work

Robustness of Vision Transformer The robustness of ViT have achieved great attention due to its great success in many vision tasks (Bhojanapalli et al., 2021; Naseer et al., 2021). On the one hand, (Bhojanapalli et al., 2021; Paul & Chen, 2021) show that vision transformers are more robust to natural corruptions (Hendrycks & Dietterich, 2019) compared to CNNs. On the other hand, (Shao et al., 2021) demonstrates that ViT achieves higher adversarial robustness than CNNs under adversarial attacks. These existing work, however, mainly focus on investigating the robustness of ViT when a whole image is naturally corrupted or adversarially perturbed. Instead, our work focuses on the patch-based architecture trait of ViT and study the robustness of ViT to patch-based natural corruption and adversarial perturbation.

Adversarial Patch Attack The work (Papernot et al., 2016) shows that adversarial examples can be created by perturbing only a small amount of input pixels. Further, (Brown et al., 2017) successfully create universal, robust and targeted adversarial patches. These adversarial patches therein are often placed on the main object in the images. (Karmon et al., 2018) proposes a strong adversarial patch attack method. They show that the created adversarial patches do not have to cover any main object and can be placed at image corners. In this work, we apply the adversarial patch attack in (Karmon et al., 2018) to ViT and place adversarial patches aligned with image patches.

Model	Pretraining	DataAug	Input Size	WeightStand	GroupNorm	Weight Decay
ResNet (He et al., 2016)	N	N	224	N	N	Y
BiT (Kolesnikov et al., 2020)	Y	N	480	Y	Y	N
ViT (Dosovitskiy et al., 2020)	Y	N	224/384	N	N	N
DeiT (Touvron et al., 2021)	N	Y	224/384	N	N	N

Table 1: Comparison of popular ResNet and ViT models: The difference in model robustness can not be blindly attributed to the model architectures. It can be caused by different training settings.

Model	Model Size	Clean Accu	Model	Model Size	Clean Accu
ResNet50	25M	78.79	ResNet18	12M	69.39
DeiT-small	22M	79.85	DeiT-tiny	5M	72.18

Table 2: Fair base models: DeiT and counter-part ResNet are trained with the exact same setting. Two models of each pair achieve similar clean accuracy with comparable model sizes.

3 Experimental Setting to Compare ViT and ResNet

Background Given an input image $\boldsymbol{x} \in \mathbb{R}^{H\times W\times C}$, ResNet (He et al., 2016) composed of a set of residual blocks takes $\boldsymbol{x}$ as input. The extracted feature maps in the l-th block is $\boldsymbol{x}^l \in \mathbb{R}^{H^l\times W^l\times C^l}$ where H^l, W^l, C^l are the height, the width and the number of channels of feature maps. The final feature maps are flattened and mapped into the output. Different from ResNet, ViT (Dosovitskiy et al., 2020) first reshapes the input $\boldsymbol{x}$ into a sequence of image patches $\{\boldsymbol{x}_i \in \mathbb{R}^{(\frac{H}{P}\cdot\frac{W}{P})\times(P^2\cdot C)}\}_{i=1}^N$ where P is the patch size and N is the number of patches. A class-token patch is concatenated to the patch sequence. A set of self-attention blocks is applied to obtain patch embeddings of the l-th block $\{\boldsymbol{x}_i^l\}_{i=1}^N$. The class-token patch embedding of the last block is mapped to the output.

Fair Base Models We list the state-of-the-art ResNet and ViT models and part of their training settings in Tab. 1. The techniques applied to boost different models are different, e.g. pretraining. Previous work (Hendrycks et al., 2019; Chen et al., 2020) have shown that weight decay, data augmentation and pre-training can affect model robustness. In addition, our investigation finds weight standardization and group normalization have a significant impact on model robustness (See details in Appendix A). This indicates that the difference in model robustness can not be blindly attributed to the model architectures if models are trained with different settings. Hence, we build fair base models to compare ViT and ResNet as follows.

First, we follow (Touvron et al., 2021) to choose two pairs of fair model architectures, DeiT-small vs. ResNet50 and DeiT-tiny vs. ResNet18. The two models of each pair (i.e. DeiT and its counter-part ResNet) are of similar model sizes. Further, we train ResNet50 and ResNet18 using the **exactly same setting** as DeiT-small and Deit-tiny in (Touvron et al., 2021). In this way, we make sure the two compared models, e.g., DeiT-samll and ResNet50, have similar model size, use the same training techniques, and achieve similar test accuracy (See Tab. 2). The two fair base model pairs are used across this paper for a fair comparison.

Adversarial Patch Attack We now introduce adversarial patch attack (Karmon et al., 2018) used in our study. The first step is to specify a patch position and replace the original pixel values of the patch with random initialized noise δ. The second step is to update the noise to minimize the probability of ground-truth class, i.e. maximize the cross-entropy loss via multi-step gradient ascent (Madry et al., 2017). The adversary patches are specified to align with input patches of DeiT.

Evaluation Metric We use the standard metric **Fooling Rate (FR)** to evaluate the model robustness. First, we collect a set of images that are correctly classified by both models that we compare. The number of these collected images is denoted as P. When these images are perturbed with natural patch corruption or adversarial patch attack, we use Q to denoted the number of images that are misclassified by the model. The Fooling Rate is then defined as FR = $\frac{Q}{P}$. The lower the FR is, the more robust the model is.

Model	the Number of Naturally Corrupted Patches				the Number of Adversarial Patches			
	32	96	160	196	1	2	3	4
ResNet50	3.7	18.2	43.4	49.8	**30.6**	**59.3**	**77.1**	**87.2**
DeiT-small	**1.8**	**7.4**	**22.1**	**38.9**	61.5	95.4	99.9	100
ResNet18	6.8	31.6	56.4	61.3	**39.4**	**73.8**	**90.0**	**96.1**
DeiT-tiny	**6.4**	**14.6**	**35.8**	**55.9**	63.3	95.8	99.9	100

Table 3: Fooling Rates (in %) are reported. DeiT is more robust to naturally corrupted patches than ResNet, while it is significantly more vulnerable than ResNet against adversarial patches.

4 Robustness of ViT to Patch-wise Perturbations

Following the setting in (Touvron et al., 2021), we train the models DeiT-small, ResNet50, DeiT-tiny, and ResNet18 on ImageNet 1k training data respectively. Note that no distillation is applied. The input size for training is $H = W = 224$, and the patch size is set to 16. Namely, there are 196 image patches totally in each image. We report the clean accuracy in Tab. 2 where DeiT and its counter-part ResNet show similar accuracy on clean images.

4.1 Patch-wise Natural Corruption

First, we investigate the robustness of DeiT and ResNet to patch-based natural corruptions. Specifically, we randomly select 10k test images from ImageNet-1k validation dataset (Deng et al., 2009) that are correctly classified by both DeiT and ResNet. Then for each image, we randomly sample n input image patches $\boldsymbol{x}_i$ from 196 patches and perturb them with natural corruptions. As in (Hendrycks & Dietterich, 2019), 15 types of natural corruptions with the highest level are applied to the selected patches, respectively. The fooling rate of the patch-based natural corruption is computed over all the test images and all corruption types. We test DeiT and ResNet with the same naturally corrupted images for a fair comparison.

We find that both DeiT and ResNet hardly degrade their performance when a small number of patches are corrupted (e.g., 4). When we increase the number of patches, the difference between two architectures emerges: DeiT achieves a lower FR compared to its counter-part ResNet (See Tab. 3). This indicates that DeiT is more robust against naturally corrupted patches than ResNet. The same conclusion holds under the extreme case when the number of patches $n = 196$. That is: the whole image is perturbed with natural corruptions. This is aligned with the observation in the existing work (Bhojanapalli et al., 2021) that vision transformers are more robust to ResNet under distributional shifts. More details on different corruption types can be found in Appendix B.

In addition, we also increase the patch size of the perturbed patches, e.g., if the patch size of the corrupted patch is 32×32, it means that it covers 4 continuous and independent input patches as the input patch size is 16×16. As shown in Fig. 2 (Left), even when the patch size of the perturbed patches becomes larger, DeiT (marked with red lines) is still more robust than its counter-part ResNet (marked with blue lines) to natural patch corruption.

4.2 Patch-wise Adversarial Attack

In this section, we follow (Karmon et al., 2018) to generate adversarial patch attack and then compare the robustness of DeiT and ResNet against adversarial patch attack. We first randomly select 100 images from ImageNet-1k validation set that are correctly classified by both models we compare. Following (Karmon et al., 2018), the ℓ_∞-norm bound, the step size, and the attack iterations are set to 255/255, 2/255, and 10K respectively. The averaged FR is reported.

As shown in Tab. 3, DeiT achieves much higher fooling rate than ResNet when one of the input image patches is perturbed with adversarial perturbation. This consistently holds even when we increase the number of adversarial patches, sufficiently supports that DeiT is more vunerable than ResNet against patch-wise adversarial perturbation. When more than 4 patches ($\sim$2% area of the input image) are attacked, both DeiT and ResNet can be successfully fooled with almost 100% FR.

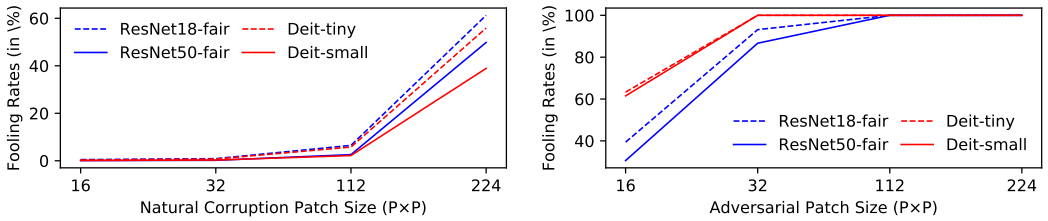


Figure 2: DeiT with red lines shows a smaller FR to natural patch corruption and a larger FR to adversarial patch of different sizes than counter-part ResNet.

When we attack a large continuous area of the input image by increasing the patch size of adversarial patches, the FR on DeiT is still much larger than counter-part ResNet until both models are fully fooled with 100% fooling rate. As shown in Fig. 2 (Right), DeiT (marked with red lines) consistently has higher FR than ResNet under different adversarial patch sizes.

Taking above results together, we discover that DeiT is more robust to natural patch corruption than ResNet, whereas it is significantly more vulnerable to adversarial patch perturbation.

5 Understanding the Robustness of ViT to Patch-wise Perturbations

In this section, we visualize and analyze models' attention to understand the different robustness performance of DeiT and ResNet against patch-wise perturbations. Although there are many existing methods, e.g., (Selvaraju et al., 2017), designed for CNNs to generate saliency maps, it is not clear yet how suitable to generalize them to vision transformers. Therefore, we follow (Karmon et al., 2018) to choose the **model-agnostic** gradient visualization method to compare the gradient (saliency) map (Zeiler & Fergus, 2014) of DeiT and ResNet when they are attacked by adversarial patches. Specifically, we first obtain the gradients of input examples towards the predicted classes, sum the absolute values of the gradients over three input channels, and visualize them.

Qualitative Evaluation As shown in Fig. 3 (a), when we use adversarial patch to attack a ResNet model, the gradient maps of the original images and the images with adversarial patch are similar. This is consistent with the observation in the previous work (Karmon et al., 2018). On the contrary, the adversarial patch can change the gradient map of DeiT by attracting more attention. As shown in Figure 3 (b), even though the main attention of DeiT is still on the object, part of the attention is misled to the adversarial patch. More visualizations are in Appendix C.

(a) on ResNet50 under Adversary Patch Attack

(b) on DeiT-small under Adversary Patch Attack

Figure 3: Gradient Visualization: the clean image, the images with adversarial patches, and their corresponding gradient maps are visualized. We use a blue box on the gradient map to mark the location of the adversarial patch. The adversary patch on DeiT attracts attention, while the one on ResNet hardly do.

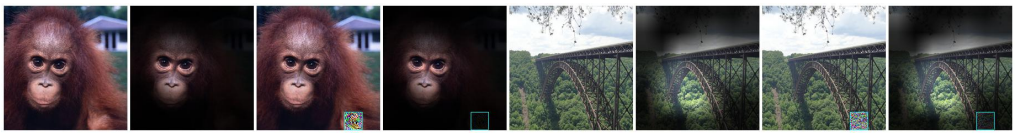
Quantitative Evaluation We also measure our observation on the attention changes with the metrics in (Karmon et al., 2018). In each gradient map, we score each patch according to (1) the maximum absolute value within the patch (MAX); and (2) the sum of the absolute values within the patch (SUM). We first report the percentage of patches where the MAX is also the maximum of the whole gradient map. Then, we divide the SUM of the patch by the SUM of the all gradient values and report the percentage.

As reported in Tab. 4, the pixel with the maximum gradient value is more likely to fall inside the adversarial patch on DeiT, compared to that on ResNet. Similar behaviors can be observed in the metric of SUM. The quantitative experiment also supports our claims above that adversarial patches mislead DeiT by attracting more attention.

	Towards ground-truth Class				Towards misclassified Class			
	SUM		MAX		SUM		MAX	
Patch Size	16	32	16	32	16	32	16	32
ResNet50	0.42	1.40	0.17	0.26	0.55	2.08	0.25	0.61
DeiT-small	**1.98**	**5.33**	**8.3**	**8.39**	**2.21**	**6.31**	**9.63**	**12.53**
ResNet18	0.24	0.74	0.01	0.02	0.38	1.31	0.05	0.13
DeiT-tiny	**1.04**	**3.97**	**3.67**	**5.90**	**1.33**	**4.97**	**6.49**	**10.16**

Table 4: Quantitative Evaluation: Each cell lists the percent of patches in which the maximum gradient value inside the patches is also the maximum of whole gradient map. SUM corresponds to the sum of element values inside patch divided by the sum of values in the whole gradient map. The average over all patches is reported.

Besides the gradient analysis, another popular tool used to visualize ViT is Attention Rollout (Abnar & Zuidema, 2020). To further confirm our claims above, we also visualize DeiT with Attention Rollout in Fig. 4. The rollout attention also shows that the attention of DeiT is attracted by adversarial patches. The attention rollout is not applicable to ResNet. As an extra check, we visualize and compare the feature maps of classifications on ResNet. The average of feature maps along the channel dimension is visualized as a mask on the original image. The visualization also supports the claims above. More visualizations are in Appendix D. Both qualitative and quantitative analysis above verifies our claims that the adversarial patch can mislead the attention of DeiT by attactting it.



(a) Attention on ResNet18 under Adversary Patch Attack

(b) Attention on DeiT-tiny under Adversary Patch Attack

Figure 4: Attention Comparison between ResNet and DeiT under Patch Attack: the clean image, the adversarial images, and their corresponding attention are visualized. The adversary patch on DeiT attract attention, while the ones on ResNet hardly do.

However, the gradient analysis is not available to compare ViT and ResNet on images with natural corrupted patches. When a small number of patch of input images are corrupted, both Deit and ResNet are still able to classify them correctly. The slight changes are not reflected in vanilla gradients since they are noisy. When a large area of the input image is corrupted, the gradient is very noisy and semantically not meaningful. Due to the lack of a fair visualization tool to compare DeiT and ResNet on naturally corrupted images, we apply Attention Rollout to DeiT and Feature Map Attention visualization to ResNet for comparing the their attention.

(a) Attention on ResNet18 under Natural Patch Corruption

(b) Attention on DeiT-tiny under Natural Patch Corruption

Figure 5: Attention Comparison between ResNet and DeiT under Natural Patch Corruption: the clean image, the naturally corrupted images, and their corresponding attention are visualized. The patch corruptions on DeiT are ignored by attending less to the corrupted patches, while the ones on ResNet are treated as normal patches.

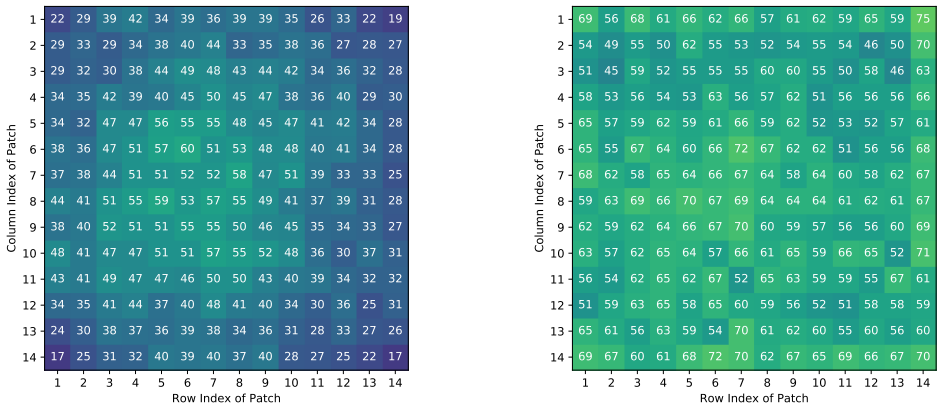
The attention visualization of these images is shown in Fig. 5. We can observe that ResNet treats the naturally corrupted patches as normal ones. The attention of ResNet on natually patch-corrupted images is almost the same as that on the clean ones. Unlike CNNs, DeiT attends less to the corrupted patches when they cover the main object. When the corrupted patches are placed in the background, the main attention of DeiT is still kept on the main object. More figures are in Appendix D.

6 Probing the Robustness of ViT to Adversarial Patches

In this section, we further investigate the properties of adversary patches created on DeiT from the perspectives of patch positions, patch alignment, and patch attack effectiveness. Concretely, we answer the following three questions: Does DeiT show similar vulnerability in different patch positions? Is it necessary to keep adversary patch postion aligned with input patches of DeiT? Are the patch attacks still able to fool DeiT in various attack settings?

6.1 Sensitivity to Patch Positions

To investigate the sensitivity against the location of adversarial patch, we visualize the FR on each patch position in Fig. 6. We can clearly see that adversarial patch achieves higher FR when attacking DeiT-tiny than ResNet18 in different patch positions. Interestingly, we find that the FRs in different patch positions of DeiT-tiny are similar, while the ones in ResNet18 are center-clustered. A similar pattern is also found on DeiT-small and ResNet50 in Appendix E.



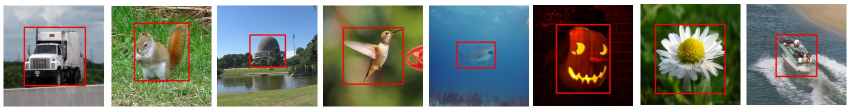
(a) Adversarial Patch Attack FRs on ResNet18

(b) Adversarial Patch Attack FRs on DeiT-tiny

Figure 6: Patch Attack FR (in %) in each patch position is visualized. FRs in different patch positions of DeiT-tiny are similar, while the ones in ResNet18 are center-clustered.

Considering that ImageNet are center-biased where the main objects are often in the center of the images, we cannot attribute the different patterns to the model architecture difference without further investigation. Hence, we design the following experiments to disentangle the two factors, i.e., model architecture and data bias. Specifically, we select two sets of correctly classified images from ImageNet 1K validation dataset. As shown in Fig. 7a, the first set contains images with corner bias where the main object(s) is in the image corners. In contrast, the second set is more center-biased where the main object(s) is exactly in the central areas, as shown in Fig. 7b.

(a) Corner-biased Images



(b) Center-biased Images

Figure 7: Collection of two sets of biased data: the fist set contains only images with corner-biased object(s), and the other set contains center-biased images.

We apply patch attack to corner-biased images (i.e., the first set) on ResNet. The FRs of patches in the center area are still significantly higher than the ones in the corner (See Appendix F). Based on this, we can conclude that such a relation of FRs to patch position on ResNet is caused by ResNet architectures instead of data bias. The reason behind this might be that pixels in the center can affect more neurons of ResNet than the ones in corners.

Similarly, we also apply patch attack to center-biased images (the second set) on DeiT. We observe that the FRs of all patch positions are still similar even the input data are highly center-biased (See Appendix F). Hence, we draw the conclusion that DeiT shows similar sensitivity to different input patches regardless of the content of the image. We conjecture it can be explained by the architecture trait of ViT, in which each patch equally interact with other patches regardless of its position.

6.2 Alignment of Adversarial Patches with Input Patches

Attack with Unaligned Patches All the previous sections study the case that the patch-wise perturbation is added onto an individual input patch $\boldsymbol{x}_i$ or an area includes multiple input patches. In other words, the adversarial patch is perfectly aligned with the input patch. In this section, we focus on different architectures' sensitivity to the alignment between adversarial patches and input patches. Specifically, we apply adversarial patch of the same size as a single input patch to a random area in the image. We find that DeiT becomes less vulnerable to adversarial patch attack, e.g., the FR on DeiT-small decreases from 61.5% to 47.9%. Intuitively, when the adversarial patch is not perfectly aligned with input patch, i.e., the attacker can only manipulate part of patch pixels rather than a full patch, the attention of DeiT is less likely to be manipulated. Similarly, we also apply an patch attack to a random image area on ResNet. As expected, the FR on ResNet stays similar (aligned 30.6 vs. unaligned 31.2) because ResNet does not process a whole image based on patches.

Therefore, we can conclude that DeiT is sensitive to the alignment between adversarial patch and input patch whereas ResNet is not due to their different architecture structures.

(Un)aligned Transfer of Adversarial Patch Perturbation (Karmon et al., 2018) shows that the adversarial patch created on an image on ResNet is not effective anymore when shifted even a single pixel away. We also conduct similar experiments on DeiT. We find that the adversarial patch perturbation on DeiT does not transfer well either when only shifted a single-pixel away. However, when shifted to match another input patch exactly, the adversarial patch is still highly effective.

Namely, the adversarial perturbation can be still effective when aligned with a different patch. The reason behind this is that, when the adversarial patch is switched to another patch, the network attention can still be misled as shown in Sec. 5. When shifted in a single pixel, the structure of

Trans-(X,Y)	(0, 1)	(0, 8)	**(0, 16)**	**(0, 32)**	(1, 0)	(0, 8)	**(16, 0)**	**(32, 0)**	(1, 1)	(8, 8)	**(16, 16)**
ResNet50	0.06	0.17	0.31	0.48	0.06	0.20	0.18	0.40	0.08	0.20	0.35
DeiT-small	0.27	0.13	**8.43**	**4.26**	0.28	0.19	**8.13**	**3.88**	0.21	0.22	**4.97**
ResNet18	0.22	0.33	0.46	0.56	0.19	0.34	0.49	0.68	0.15	0.23	0.49
DeiT-tiny	2.54	2.32	**29.15**	**18.19**	2.30	1.73	**28.37**	**17.32**	2.11	2.29	**21.23**

Table 5: Transferability of adversarial patch across different patch positions of the the image. Translation X/Y stands for the number of pixels shifted in rows or columns. When they are shifted to cover other patches exactly, adversarial patches transfer well, otherwise not.

perturbation is destroyed due to the patch split of DeiT. Additionally, We find that the adversarial patch perturbation barely transfers across images or models regardless of the alignment. Details can be found in Appendix G.

6.3 Patch Attack under different Settings

Iterations of Patch Attack In our experiment, as in (Karmon et al., 2018), the attack iteration is set to 10k. We also check how many iterations are required to attack the classification successfully. The required iterations are averaged on all patch postions of the misclassified images. The required attack iterations on DeiT-tiny is less than that on ResNet18 (65 vs. 342). The observation also holds on DeiT-small and ResNet50 (294 vs. 455). This experiment shows DeiT is more vulnerable than ResNet from another perspective.

Imperceptible Patch Attack In this work, we use unbounded local patch attacks where the pixel intensity can be set to any value in the image range $[0, 1]$. The adversarial patches are often visible, as shown in Fig. 1. In a more popular setting of adversarial attack and defense, the maximally allowed change of the input value is 8/225, in which the adversarial perturbation is imperceptible human vision. We also compare ResNet and DeiT under this setting.

In the case of a single patch attack, the attacker achieves FR of 2.9% on ResNet18 and 11.2% on DeiT-tiny. More scores and visualization of the images with imperceptible perturbation can be found in Appendix I. DeiT is still more vulnerable than ResNet when individual patches are attacked with imperceptible perturbation. When the patch size to attack is set to be the whole image size, it is exactly the same as the standard attack. We show that both ResNet and DeiT can be easily fooled When the standard attack setting is applied.

Targeted Patch Attack A targeted attack can be achieved by setting the attack objective to maximize the probability of the target class. We also compare DeiT and ResNet under the targeted attack above. In the experiment, we randomly select a target class except for the ground-truth class for each image. In the case of a single attack patch, the attacker achieves FR of 15.4% on ResNet18 and 32.3% on DeiT-tiny. Under targeted attack, DeiT is more vulnerable than ResNet. The claim also holds on the other model pair (ResNet18 7.4% vs. DeiT-small 24.9%). Visualization of the adversarial patches is in Appendix J.

7 Conclusion

Based on the patch-based architectural trait of ViT, we investigate its robustness against two types of patch-wise perturbations: natural patch corruption and adversarial patch attack. Compared to convolutional networks (e.g., ResNet), vision transformer (e.g., DeiT) is more robust to natural patch corruption, whereas it is significantly more vunerable against adversarial patch. Extensive analysis reveals that the self-attention mechanism of vision transformers can effectively ignore natural corrupted patches but be easily misled to adversarial patches to make a mistake. Further, we find that DeiT and ResNet exhibit different sensitivities against patch positions and patch alignment of adversarial patch attacks due to their different architectural structures. We believe our work can shed light on improving robustness of transformer-based models and spur future work on investigating ViT variants, e.g., MLP-Mixer (Tolstikhin et al., 2021), and the robustness of transformers to token-wise perturbations in NLP.

REFERENCES

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

Srinadh Bhojanapalli, Ayan Chakrabarti, Daniel Glasner, Daliang Li, Thomas Unterthiner, and Andreas Veit. Understanding robustness of transformers for image classification. *arXiv preprint arXiv:2103.14586*, 2021.

Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665v1*, 2017.

Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 699–708, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

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

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In *International Conference on Machine Learning*, pp. 2712–2721. PMLR, 2019.

Danny Karmon, Daniel Zoran, and Yoav Goldberg. Lavan: Localized and visible adversarial noise. In *International Conference on Machine Learning*, pp. 2507–2515. PMLR, 2018.

Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pp. 491–507. Springer, 2020.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. 2017.

Muzammal Naseer, Kanchana Ranasinghe, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Intriguing properties of vision transformers. *arXiv preprint arXiv:2105.10497*, 2021.

Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pp. 372–387. IEEE, 2016.

Sayak Paul and Pin-Yu Chen. Vision transformers are robust learners. *arXiv preprint arXiv:2105.07581*, 2021.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Rulin Shao, Zhouxing Shi, Jinfeng Yi, Pin-Yu Chen, and Cho-Jui Hsieh. On the adversarial robustness of visual transformers. *arXiv preprint arXiv:2103.15670*, 2021.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *ICLR*, 2014.

Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, et al. Mlp-mixer: An all-mlp architecture for vision. *arXiv preprint arXiv:2105.01601*, 2021.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pp. 10347–10357. PMLR, 2021.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

A The Training Setting Can Affect Model Robustness

We train ResNet18 on CIFAR10 in the standard setting (He et al., 2016). To study the impact of training settings on model robustness, we train models with different input sizes (i.e., 32, 48, 64), with or without Weight Standardization and Group Normalization to regularize the training process. The foolong rate of single patch attack is reported. Especially, with our experiments, we find that Weight Standardization and Group Normalization can have a significant impact on model robustness (See Tab. 6). The two techniques are applied in BiT (Kolesnikov et al., 2020) to improve its performance. However, they are not applied to standard ViT and DeiT training settings. Hence, the robustness difference between ViT and BiT cannot be attributed to the difference of model architectures.

Note that a comprehensive study of the relationship between all factors of training and model adversarial robustness is out of the scope of this paper. We aim to point out that these factors can have an impact on model robustness to different extents. The robustness difference cannot be blindly attributed to the difference of model architectures. We need to build new fair base models to study the robustness of ResNet and ViT.

Model	Input Size		
ResNet18	32	48	64
Clean Accuracy	93.4	93.8	93.7
FR of Patch Attack	35.9	42.2	39.2

Model	Training Techniques			
ResNet18	No	WS	GN	WS + GN
Clean Accu	93.4	93.6	92.0	93.8
Patch Attack FR	35.9	51.3	52.6	71.1

Table 6: Study of the training factors on the relation to model robustness: While the input size has minor impact on model robustness, Weight Standardization (WS) and Group Normalization (GN) can change model robustness significantly.

B Natural Patch Corruption with Different Levels and Types

Models can show different robustness when the inputs are corrupted with different natural noise types. To better evaluate the model robustness to natural corruption, the work (Hendrycks & Dietterich, 2019) summarizes 15 common natural corruption types. The averaged score is used as an indicator of model robustness. In this appendix section, we show more details of model robustness to different noise types. As show in Fig. 8 and 9, The FR on DeiT is lower than on ResNet. We conclude that DeiT is more robust than ResNet to natural patch corruption.

Furthermore, we also investigate the model robustness in terms of different noise levels. As shown in Fig. 10 and 11. The different colors stand for different noise level. S1-S5 corresponds to the natural corruption severity from 1 to 5. In each noise type, the left bar corresponds to ResNet variants and the right one to DeiT variants. We can observe that DeiT show lower FR in each severity level. Namely, the conclusion drawn above also holds across different noise levels.

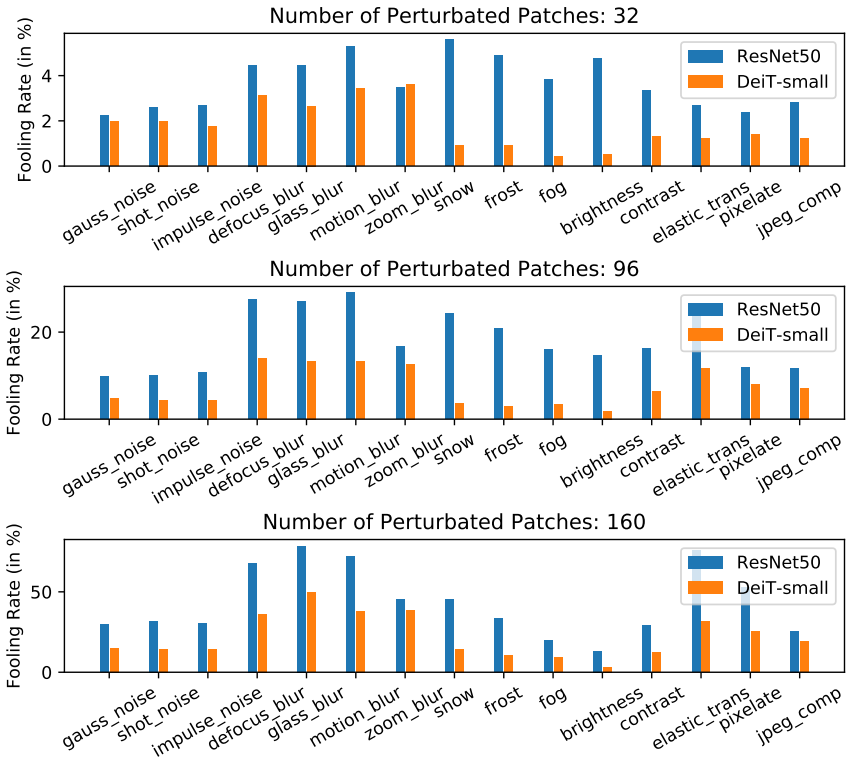


Figure 8: Comparison of ResNet50 and Deit-small on Naturally Corrupted Patches

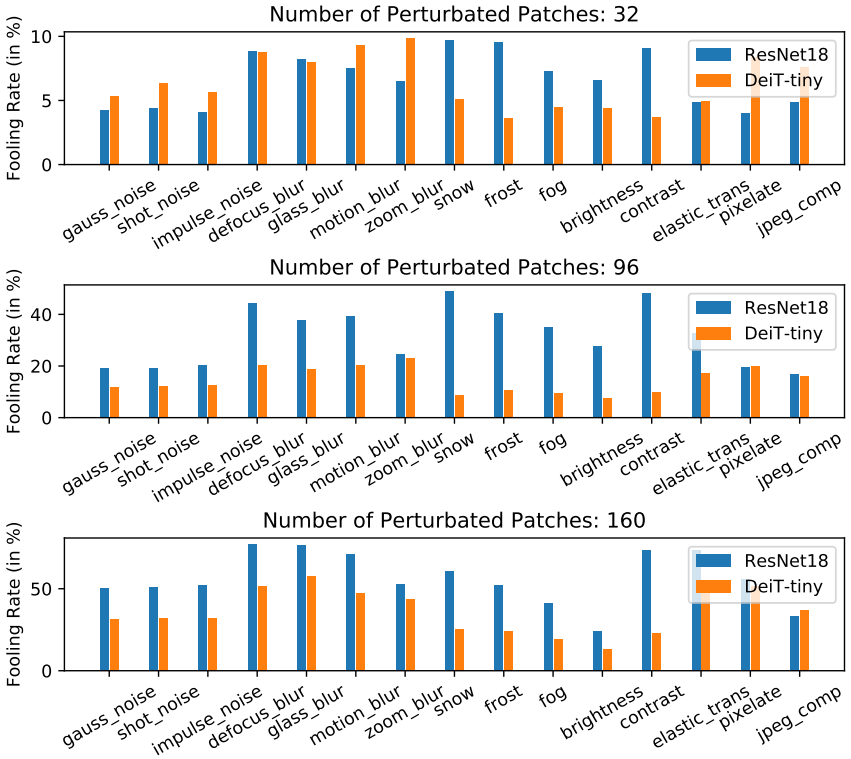


Figure 9: Comparison of ResNet18 and Deit-tiny on Naturally Corrupted Patches

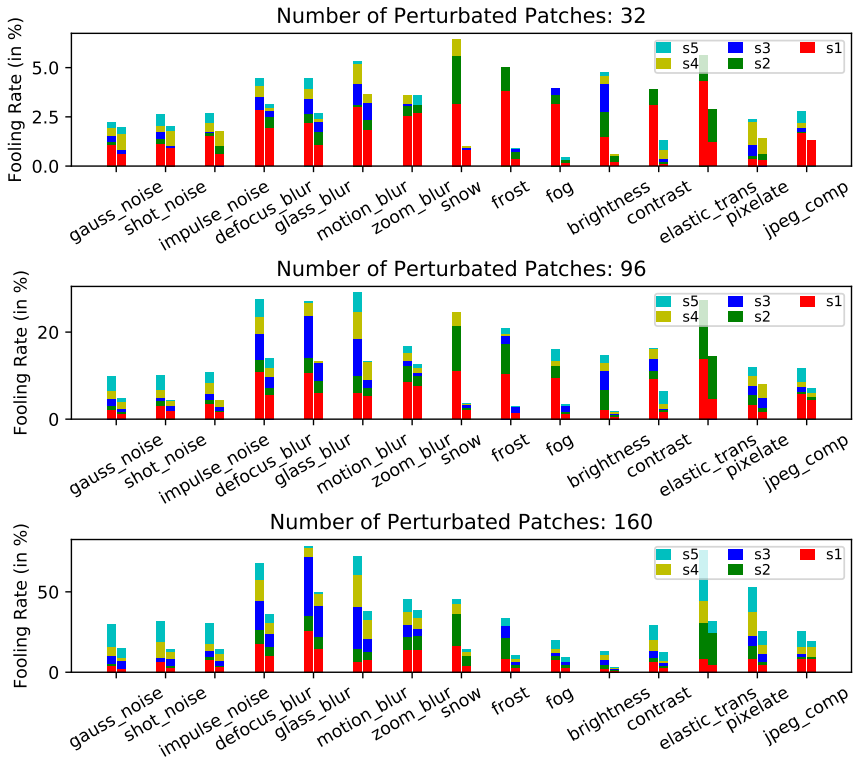


Figure 10: Comparison of ResNet50 and Deit-small on Patches Corrupted with Different Levels

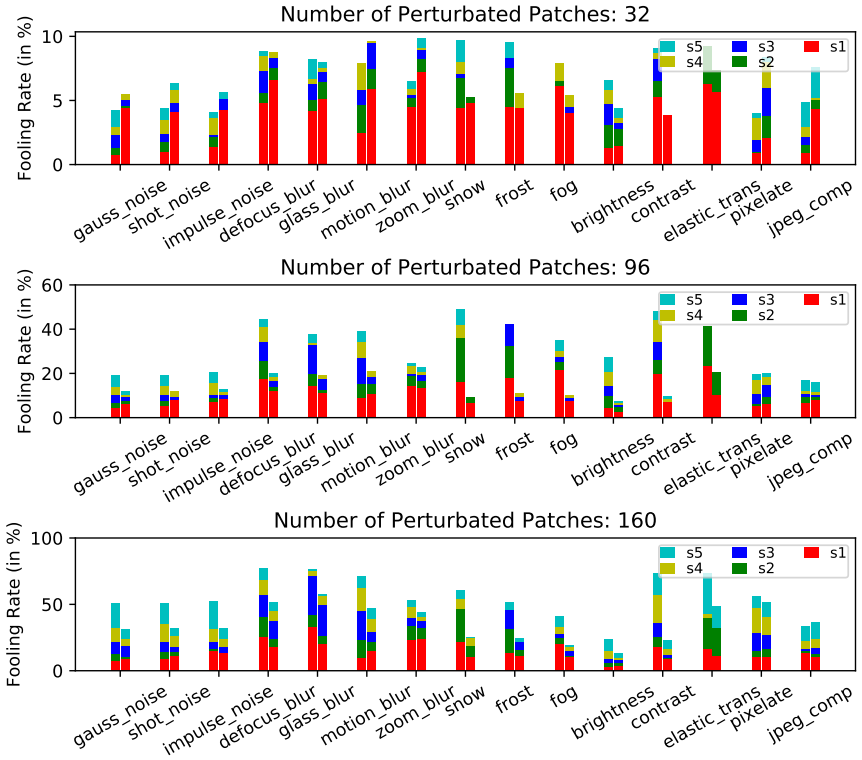


Figure 11: Comparison of ResNet18 and Deit-tiny on Patches Corrupted with Different Levels

C GRADIENT VISUALIZATION OF ADVERSARIAL IMAGES

We first get the absolute value of gradient received by input and sum them across the channel dimension. The final values are mapped into gray image scale. We also mark the adversarial patch with a blue bounding box in the visualized gradient maps.

The adversarial patch noises with different patch sizes (i.e., P=16 and P=32) are shown on DeiT and ResNet in Fig. 17, 18, 19, and 20. In each row of these figures, we fist show the clean image and visualize the gradients of inputs as a mask on the image. Then, we show the images with patch noises on different patch positions, and the gradient masks are also shown following the corresponding adversarial images.

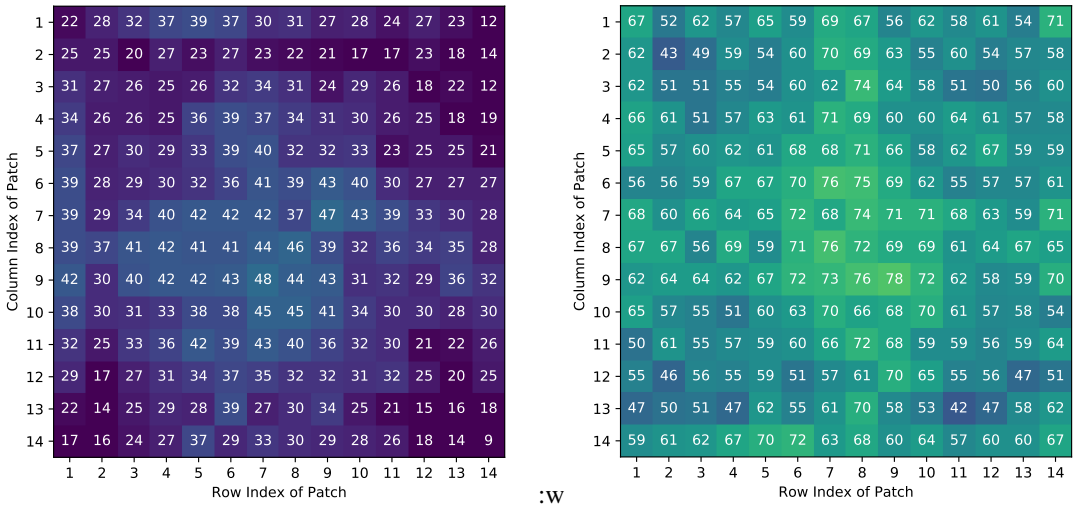
D ATTENTION ROLLOUT ON DEIT AND FEATURE MAP MASKS ON RESNET

In this appendix section, we show more Attention Rollout on DeiT and Feature Map Masks on ResNet. The adversarial patch noises with different patch sizes are shown (i.e., P=16 and P=32) in Fig. 21, 22, 23, and 24. In each row of these figures, we fist show the clean image and visualize the attention as a mask on the image. Then, we show the images with patch noises on different patch positions, and the attention masks are also shown following the correspond adversarial images.

The rollout attention on DeiT and Feature Map mask on ResNet on naturally corrupted images are shown in Fig. 25, 26, 27, and 28. We can observe that ResNet treats tha corrupted patches as normal ones. On DeiT, the attention is slightly distract by naturally corrupted patches when they are in the background. However, the main attention is still on the main object of input.

E FOOLING RATES OF EACH PATCH ON RESNET50 AND DEIT-SMALL

The FRs in different patch positions of DeiT are similar, while the ones in ResNet are center-clustered. A similar pattern can also be found on DeiT-small and ResNet50 in Fig. 12.



:w

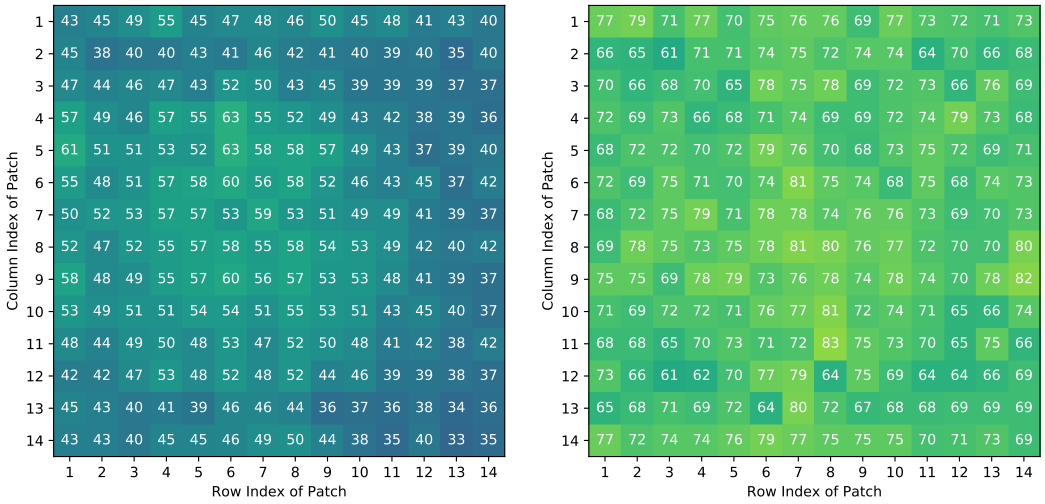
(a) Adversarial Patch Attack FRs on ResNet50

(b) Adversarial Patch Attack FRs on DeiT-small

Figure 12: Patch Attack FR (in %) in each patch position is visualized on ResNet50 and DeiT-small.

F FOOLING RATES OF EACH PATCH ON RESNET AND DEIT ON BIASED DATA

In the coner-biased image set, the FR on ResNet is still center-clustered, as shown in Fig. 13a. In the center-biased image set, the FR on DeiT is still similar on different patch postions, as shown in Fig. 13b.



(a) FRs of ResNet18 on Corner-biased Data

(b) FRs of DeiT-tiny on Center-biased Data

Figure 13: Patch Attack FR (in %) in each patch position is visualized on ResNet18 and DeiT-tiny on biased data.

G Transferability of Adversarial Patches across Images, Models, and Patch Positions

As shown in Tab. 7, the adversarial patch noise created on a given image hardly transfer to other images. When large patch size is applied, the patch noises on DeiT transfer slightly better than the ones on ResNet.

Models	ResNet50	DeiT-small	ResNet18	DeiT-tiny
across images (Patch Size=16)	3.5	2.1	3.4	6.4
across images (Patch Size=112)	8.1	13.4	10.6	21.5

Table 7: Transferability of adversarial patch across different images

The transferbility of adversrial noise between Vision Transformer and ResNet has already explored in a few works. They show that the transferability between them is remarkablely low. As shown in Tab. 8, the adversarial patch noise created on a given image does not transfer to other models.

	Patch Size=16				Patch Size=112			
Models	ResNet50	DeiT-small	ResNet18	DeiT-tiny	ResNet50	DeiT-small	ResNet18	DeiT-tiny
ResNet50	-	0.3	0.16	2.2	-	5.25	8	11.75
DeiT-small	0.04	-	0.09	1.79	5.5	-	9.25	12.25
ResNet18	0.09	0.22	-	1.9	5.75	5	-	12
DeiT-tiny	0.04	0.13	0.06	-	5.5	5	9.25	-

Table 8: Transferability of adversarial patch across models

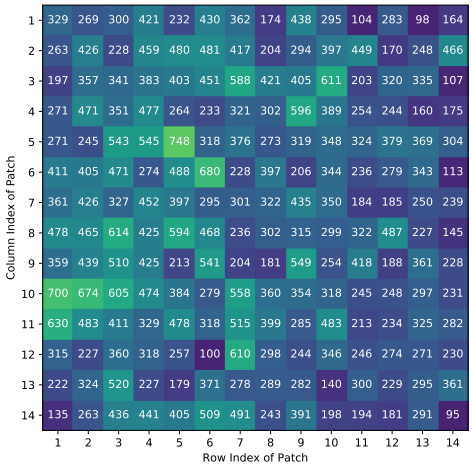
When they are transfered to another patch, the adversarial patch noises are still highly effective. However, the transferability of patch noise can be low, when the patch is not aligned with input patches. The claim on the patch noise with size of 112 is also true, as shown in Tab. 9.

Model	ResNet50	DeiT-small	ResNet18	DeiT-tiny
across positions (0, 4)	6.25	5.25	11.25	12.75
across positions (0, 16)	5.75	**34.5**	11.5	**54**
across positions (0, 64)	6	22	9.5	30.75
across positions (4, 0)	6.5	5.75	9.75	12.5
across positions (16, 0)	7.25	**35**	10.25	**54**
across positions (64, 0)	5.5	18.25	9.25	31
across positions (4, 4)	6	4.75	8.5	13.5
across positions (16, 16)	4.5	**18.5**	9	**33**
across positions (64, 64)	6	9.75	8.25	17.5

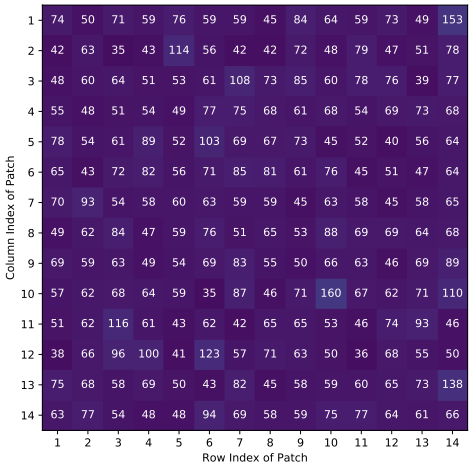
Table 9: Transferability of adversarial patch across patch positions

H CONVERGENCE SPEED IN EACH PATCH ON RESNET50 AND DEIT-SMALL

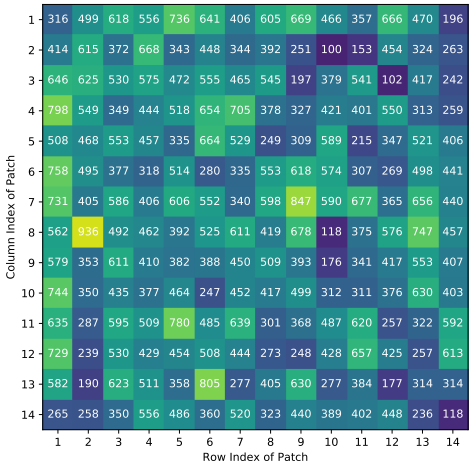
As shown in Fig. 14, the required attack iterations on DeiT is also less than counter-part ResNet.



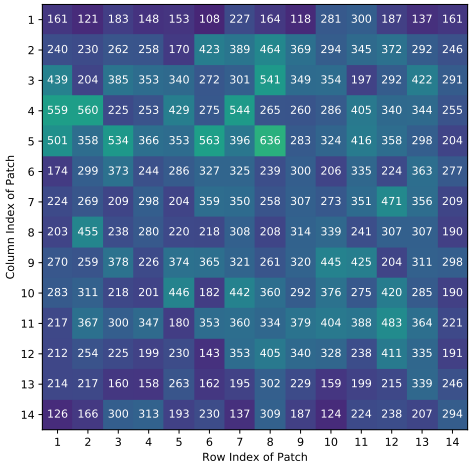
(a) Required Attack Iterations on ResNet18



(b) Required Attack Iterations on DeiT-tiny



(c) Required Attack Iterations on ResNet50



(d) Required Attack Iterations on DeiT-small

Figure 14: Convergence of Adversarial Patch Attack on Different Architectures

I MORE SETTINGS AND VISUALIZATION OF ADVERSARIAL EXAMPLES WITH IMPERCEPTIBLE NOISE

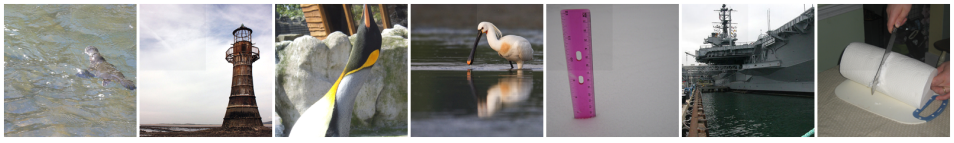
In the standard adversarial attack, the artificial noise can be placed anywhere in the image. In our adversarial patch attack, we conduct experiments with different patch sizes, which are multiple times the size of a single patch. The robust accuracy under different attack patch sizes is reported in Tab. . We can observe that DeiT is more vulnerable than ResNet under imperceptible attacks.

Model	Patch-Size=16	Patch-Size=32	Patch-Size=112	Patch-Size=224
ResNet50	2.9	20.9	98.3	100
DeiT-small	4.1	38.7	100	100
ResNet18	3.1	26.0	99.1	100
DeiT-tiny	11.2	46.8	100	100

Table 10: Adversarial Patch Attack with Imperceptible Perturbation . FRs are reported in percentage.

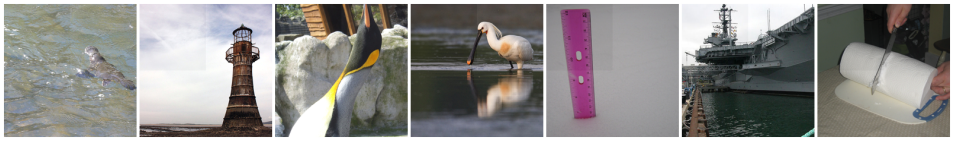
The clean images and the adversarial images created on different models are shown in Fig. 15. The adversarial examples created with imperceptible patch attack are not quasi-distinguishable from the corresponding clean images for human vision.

(a) Clean Images

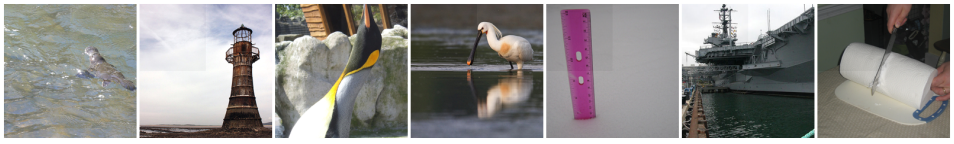


(b) Adversarial Examples on ResNet18

(c) Adversarial Examples on DeiT-tiny



(d) Adversarial Examples on ResNet50



(e) Adversarial Examples on DeiT-small

Figure 15: Visualization of Adversarial Examples with Imperceptible Patch Noise: The adversarial images with patch noise of size 112 in the left-upper corner of the image are visualized. Please Zoom in to find the subtle difference.

J VISUALIZATION OF ADVERSARIAL PATCH NOISE

Besides reporting the FRs, we also visualize the adversarial patch noises created on ResNet and DeiT. The adversarial patch noise created with Equation (**??**) are shown in Fig. 16a and 16c. We are not able to recognize any object in the target class.

Following (Karmon et al., 2018), we enhance the attack algorithm where we place the patch noise on different patch positions in different images in each attack iteration. From the visualization of the created noise in Fig. 16b and 16d, we can recognize the object/object parts of the target class on both ResNet and DeiT. In this appendix section, we conclude that the recognizability of adversarial patch noise is dependent more on attack algorithms than the model architectures.

(a) Patch Noise on ResNet50 under the 1st Setting

(b) Patch Noise on ResNet50 under the 2nd Setting

(c) Patch Noise on DeiT-small under the 1st Setting

(d) Patch Noise on DeiT-small under the 2nd Setting

Figure 16: Visualization of Adversarial Patch Perturbations under different Settings: In the 1st setting, the patch noise is created to fool a single classification in a given patch position. The goal in the 2nd setting to mislead the classifications of a set of images at all patch positions.

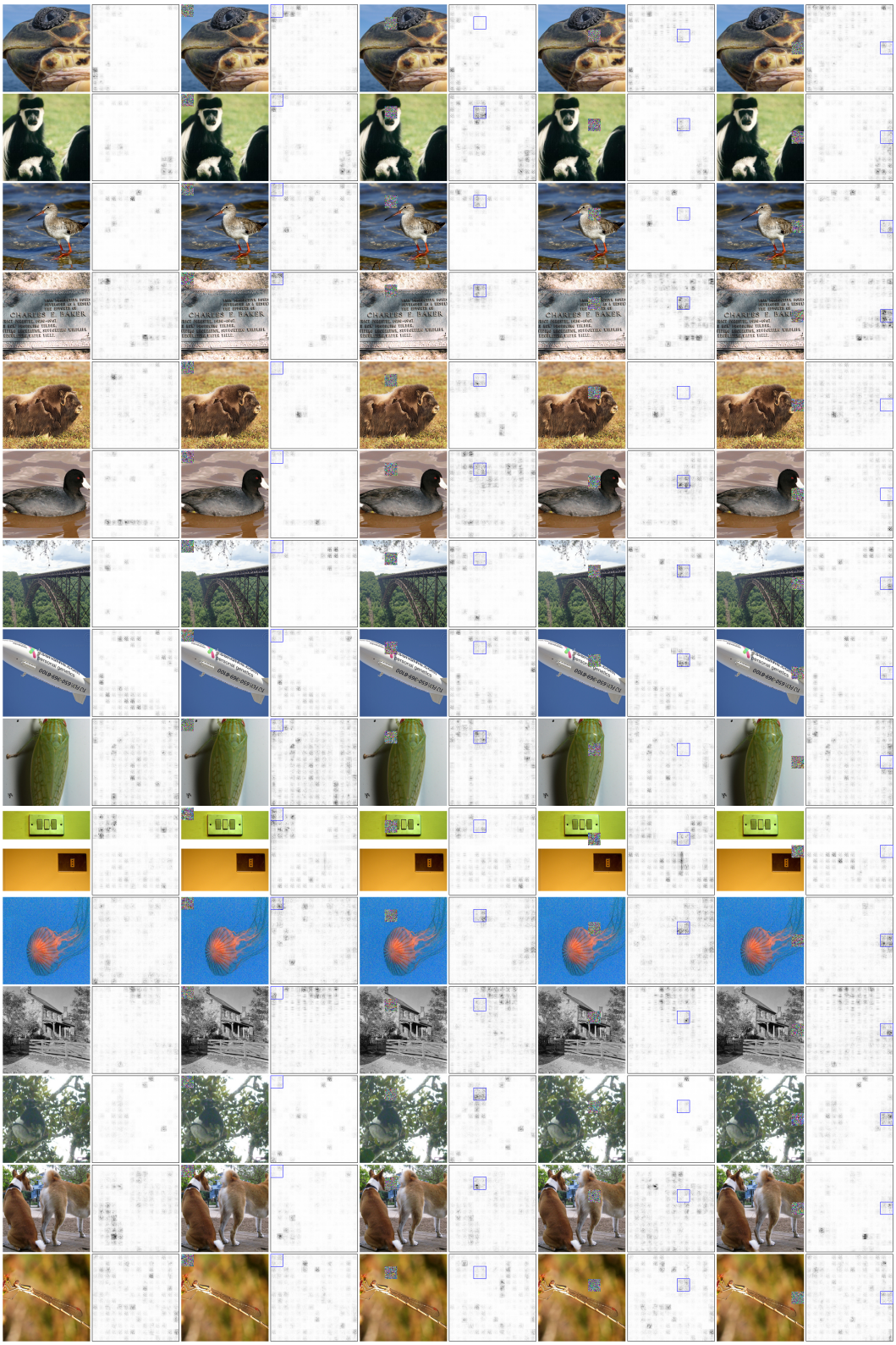


Figure 17: Gradient Visualization on DeiT-small with Attack Patch size of 32

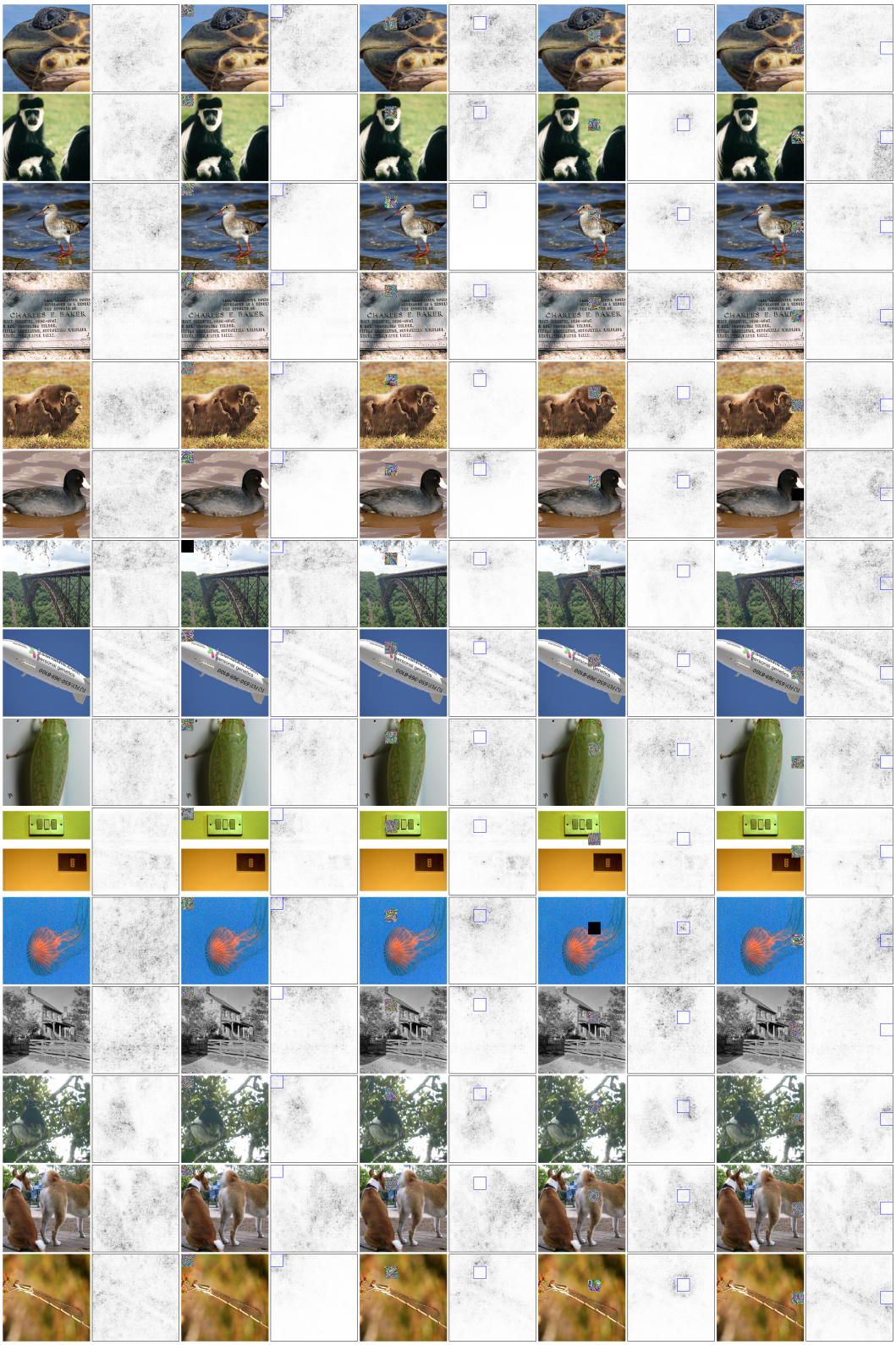


Figure 18: Gradient Visualization on ResNet50 with Attack Patch size of 32

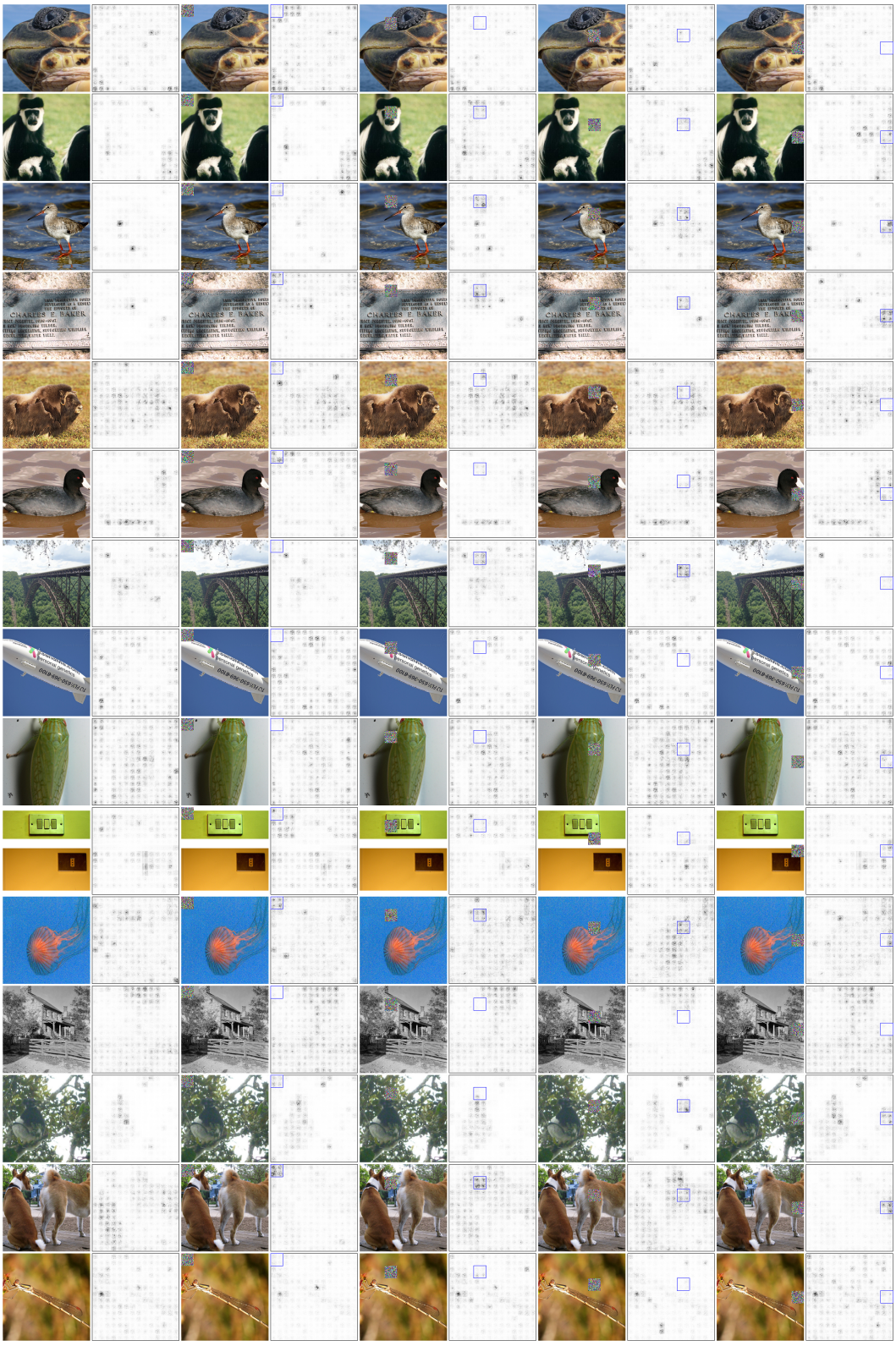


Figure 19: Gradient Visualization on DeiT-tiny with Attack Patch size of 32

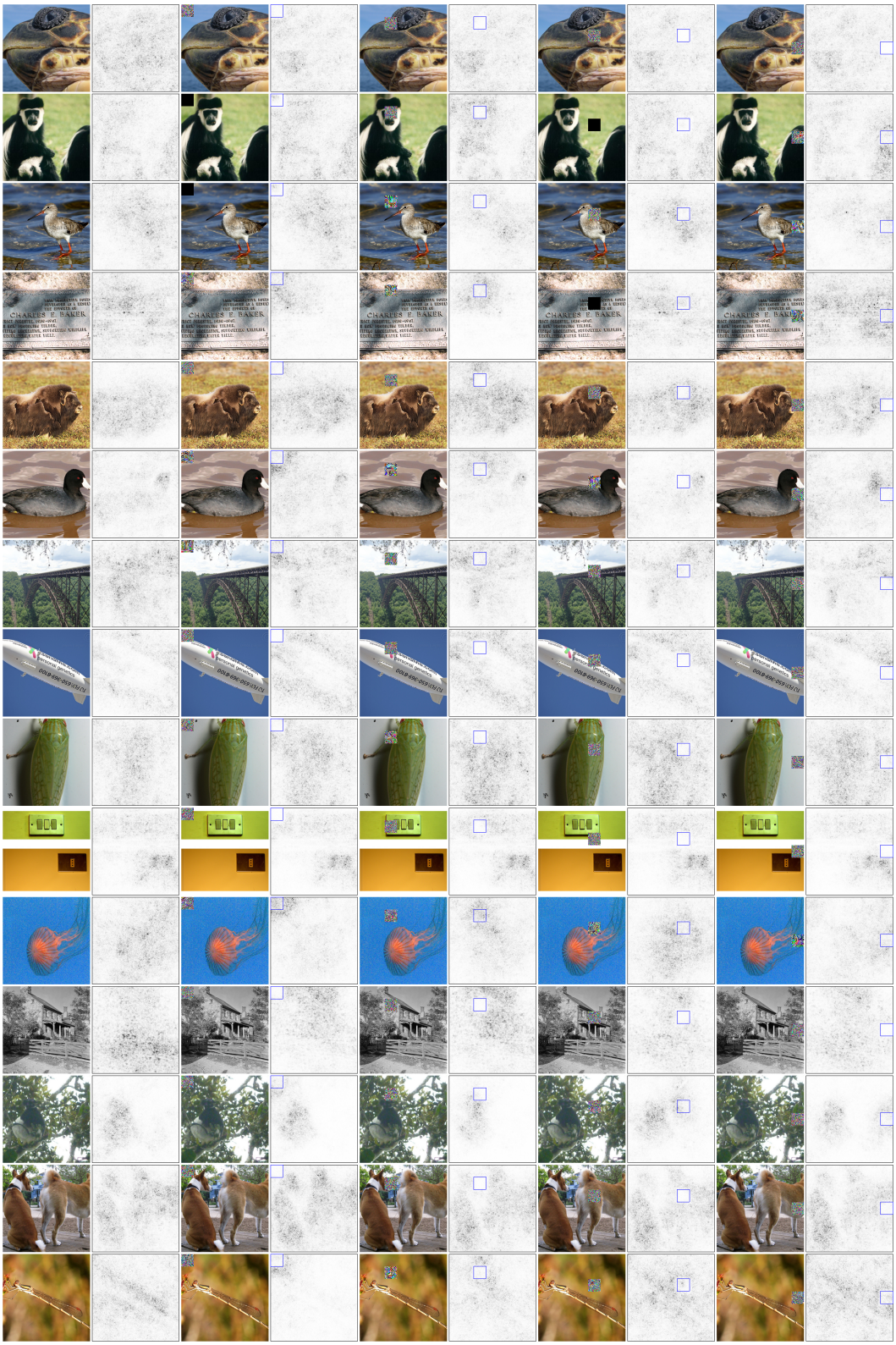


Figure 20: Gradient Visualization on ResNet18 with Attack Patch size of 32

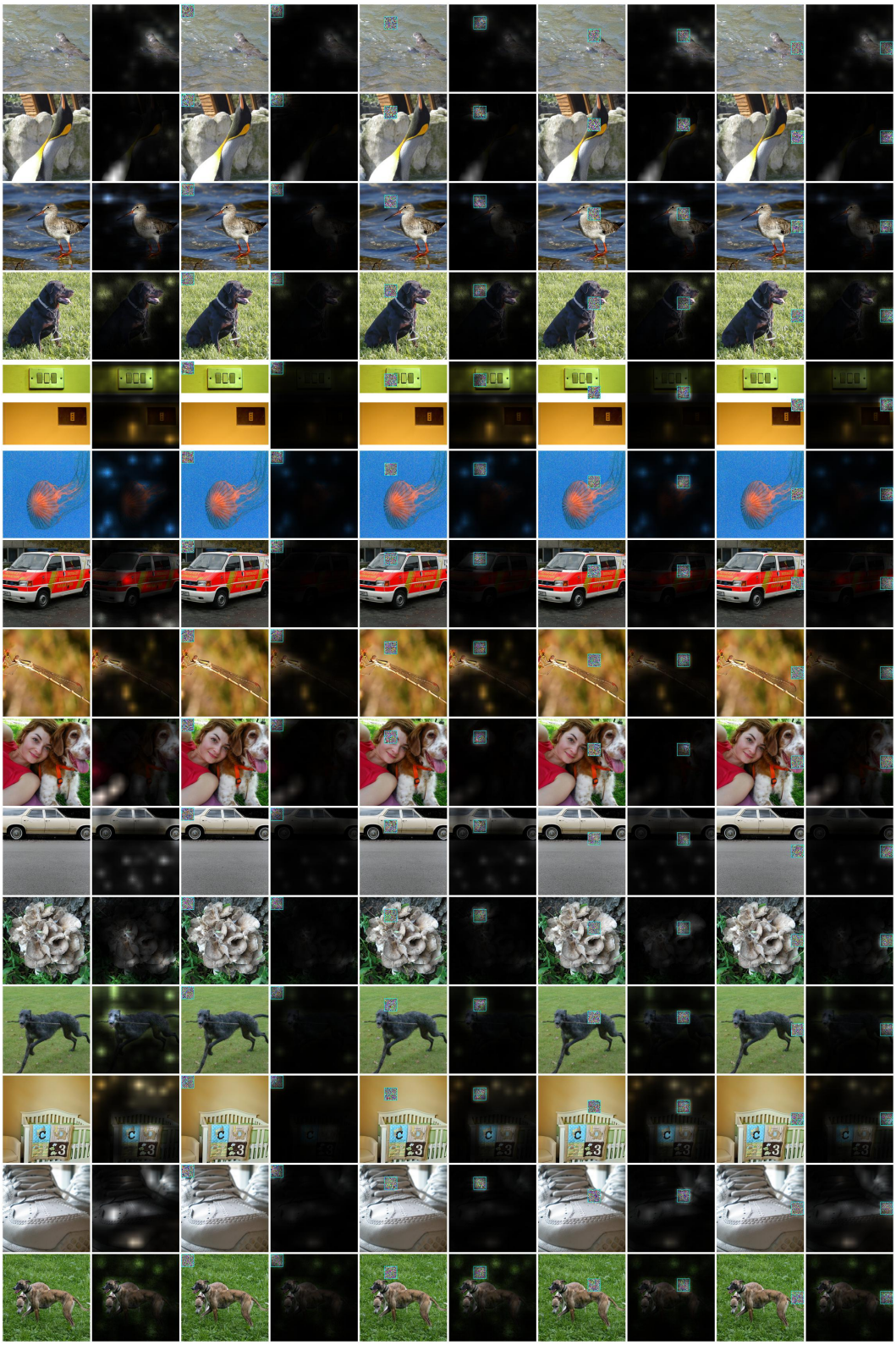


Figure 21: Rollout Attention on DeiT-small with Attack Patch size of 32 on Adversarial Images

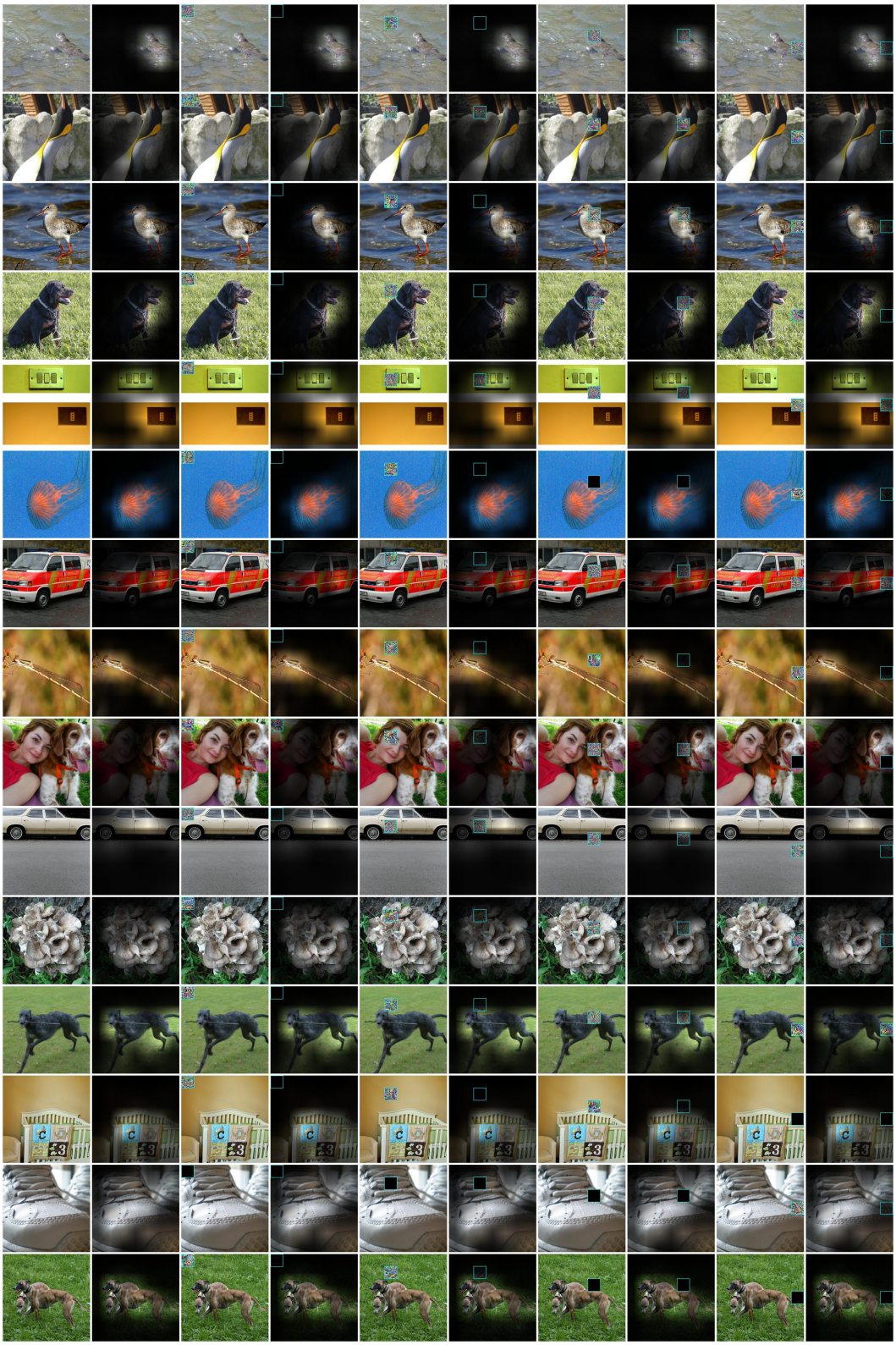


Figure 22: Averaged Feature Maps of ResNet50 as Attention with Attack Patch size of 32 on Adversarial Images

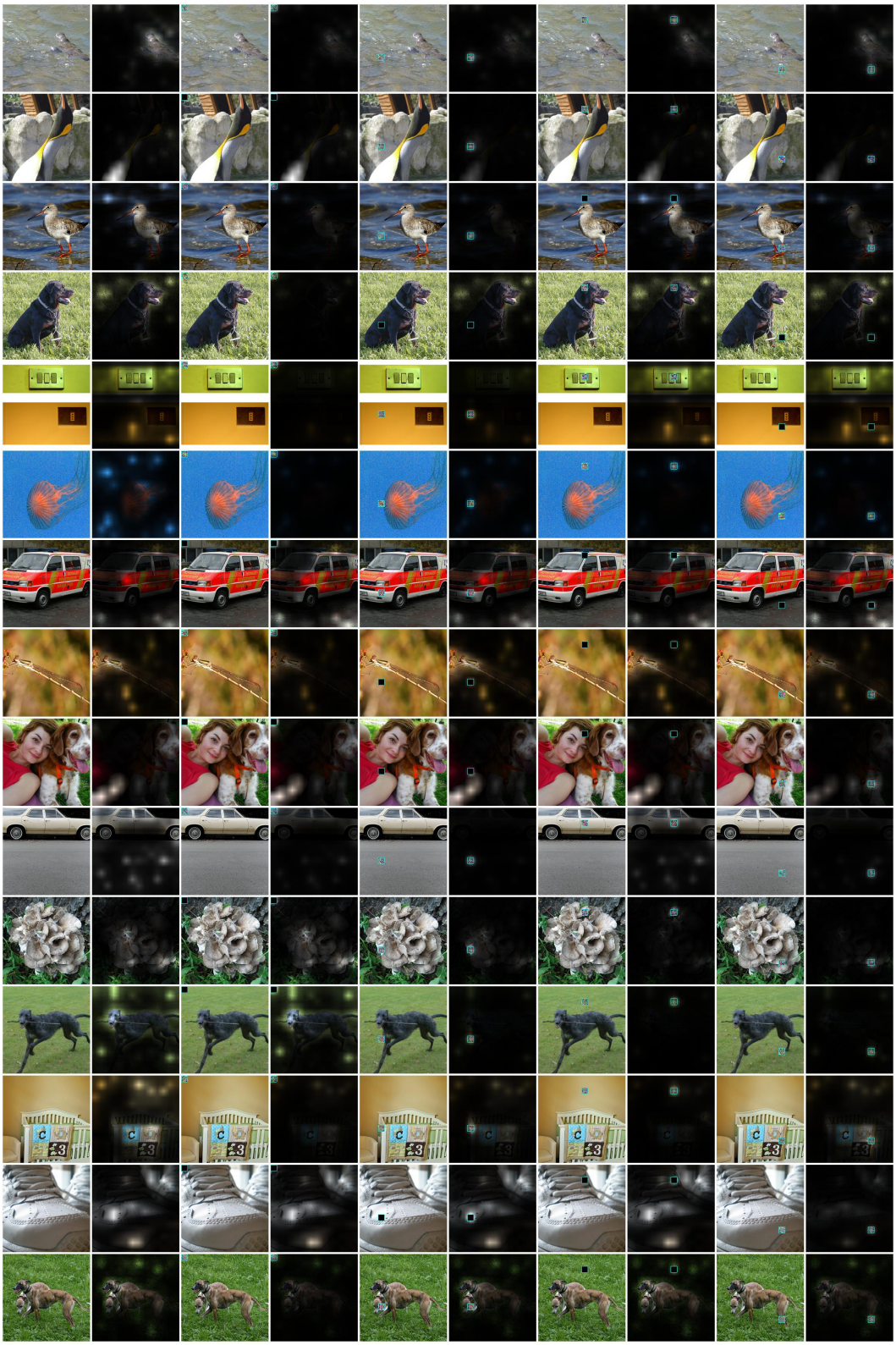


Figure 23: Rollout Attention on DeiT-small with Attack Patch size of 16 on Adversarial Images

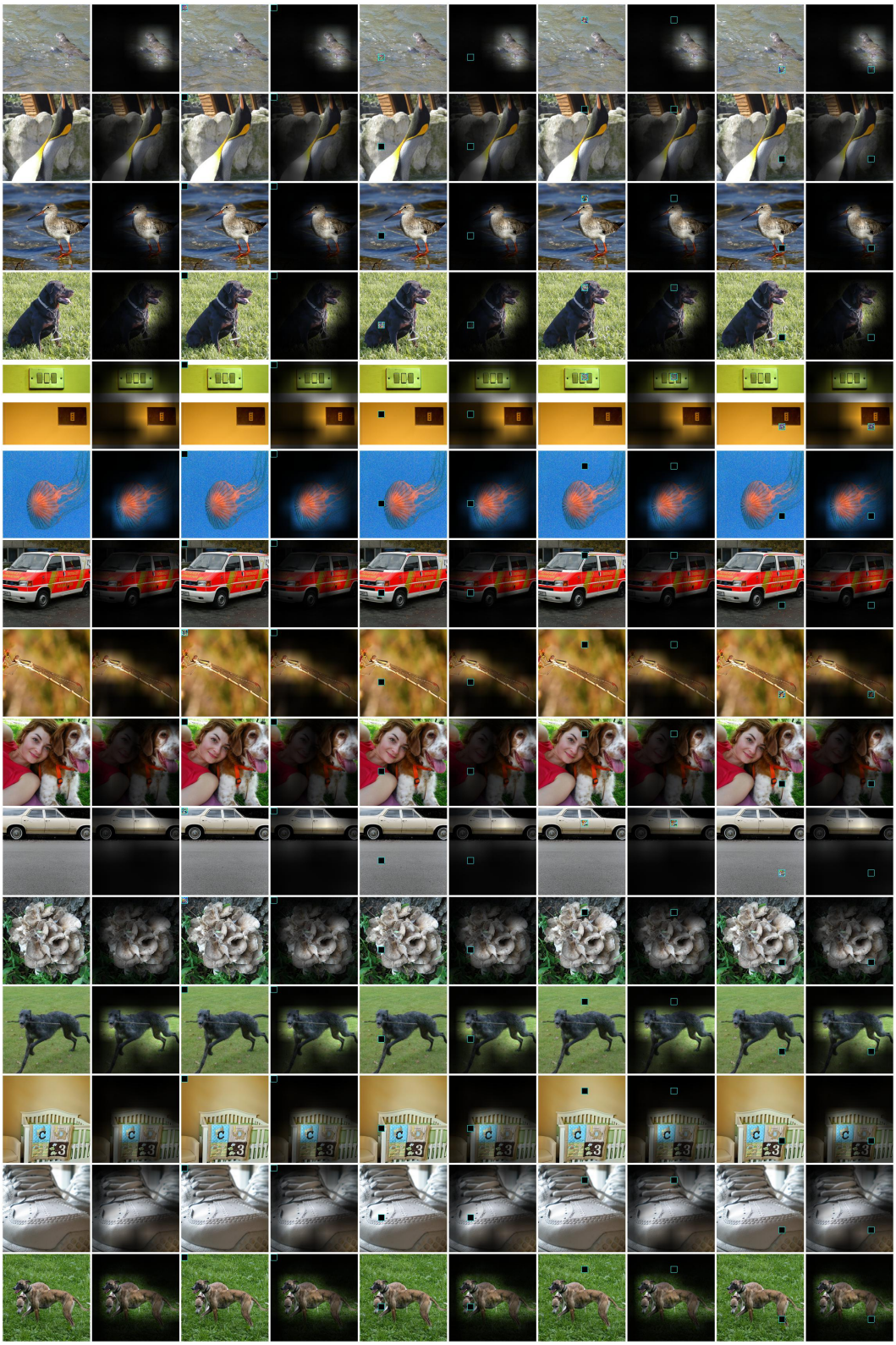


Figure 24: Averaged Feature Maps of ResNet50 as Attention with Attack Patch size of 16 on Adversarial Images

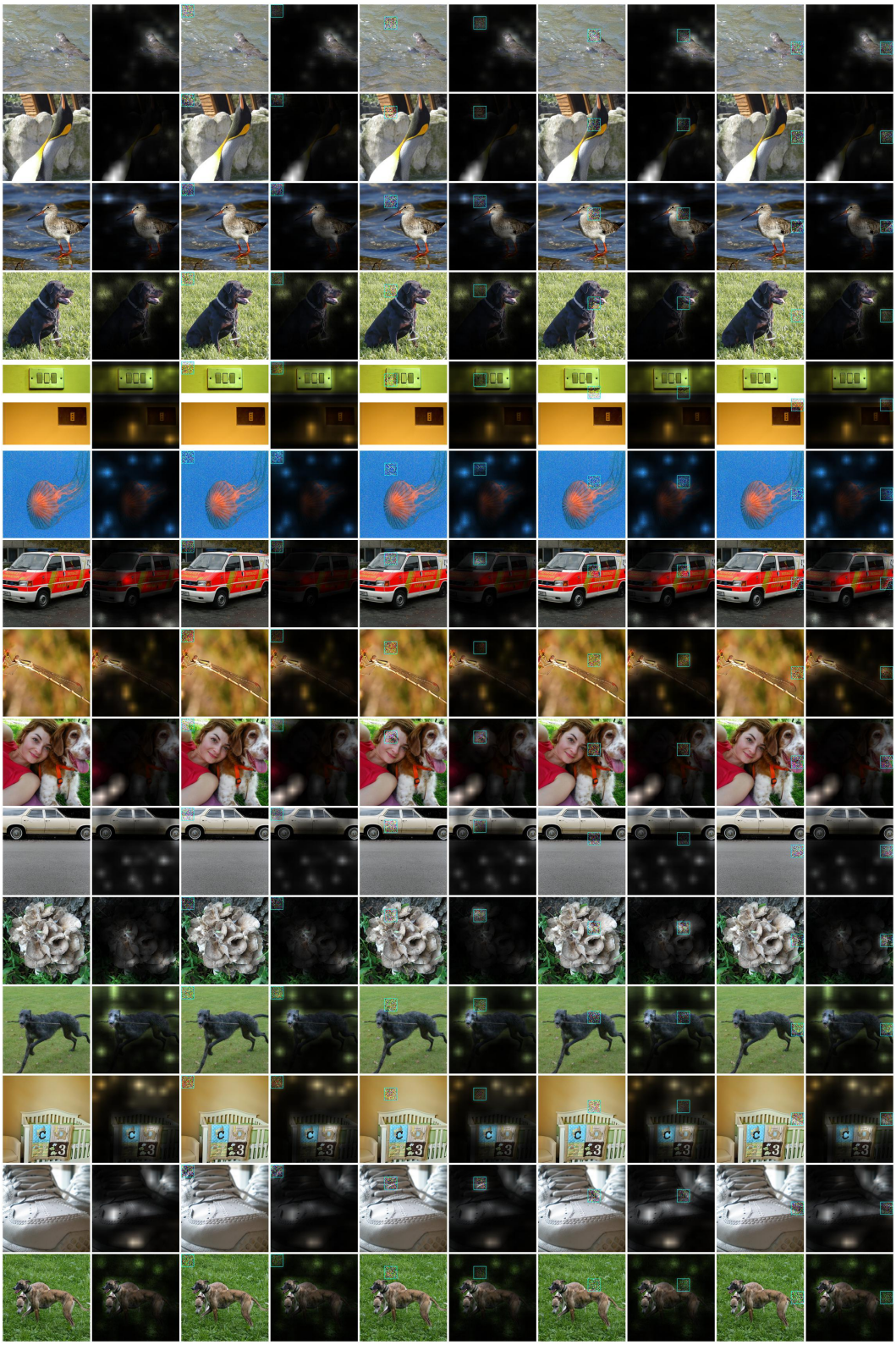


Figure 25: Rollout Attention on DeiT-small with Attack Patch size of 32 on Corrupted Images

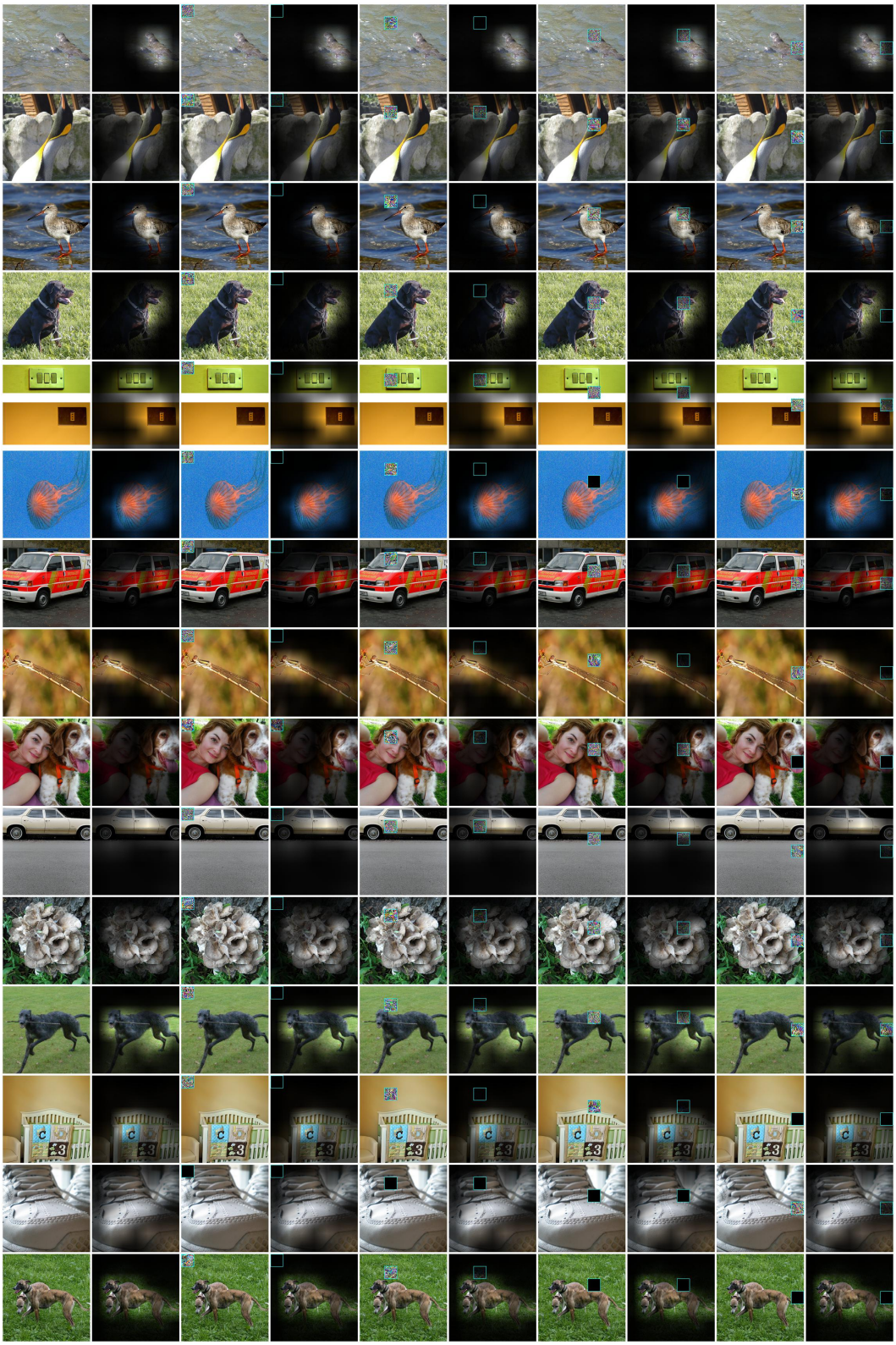


Figure 26: Averaged Feature Maps of ResNet50 as Attention with Attack Patch size of 32 on Corrupted Images

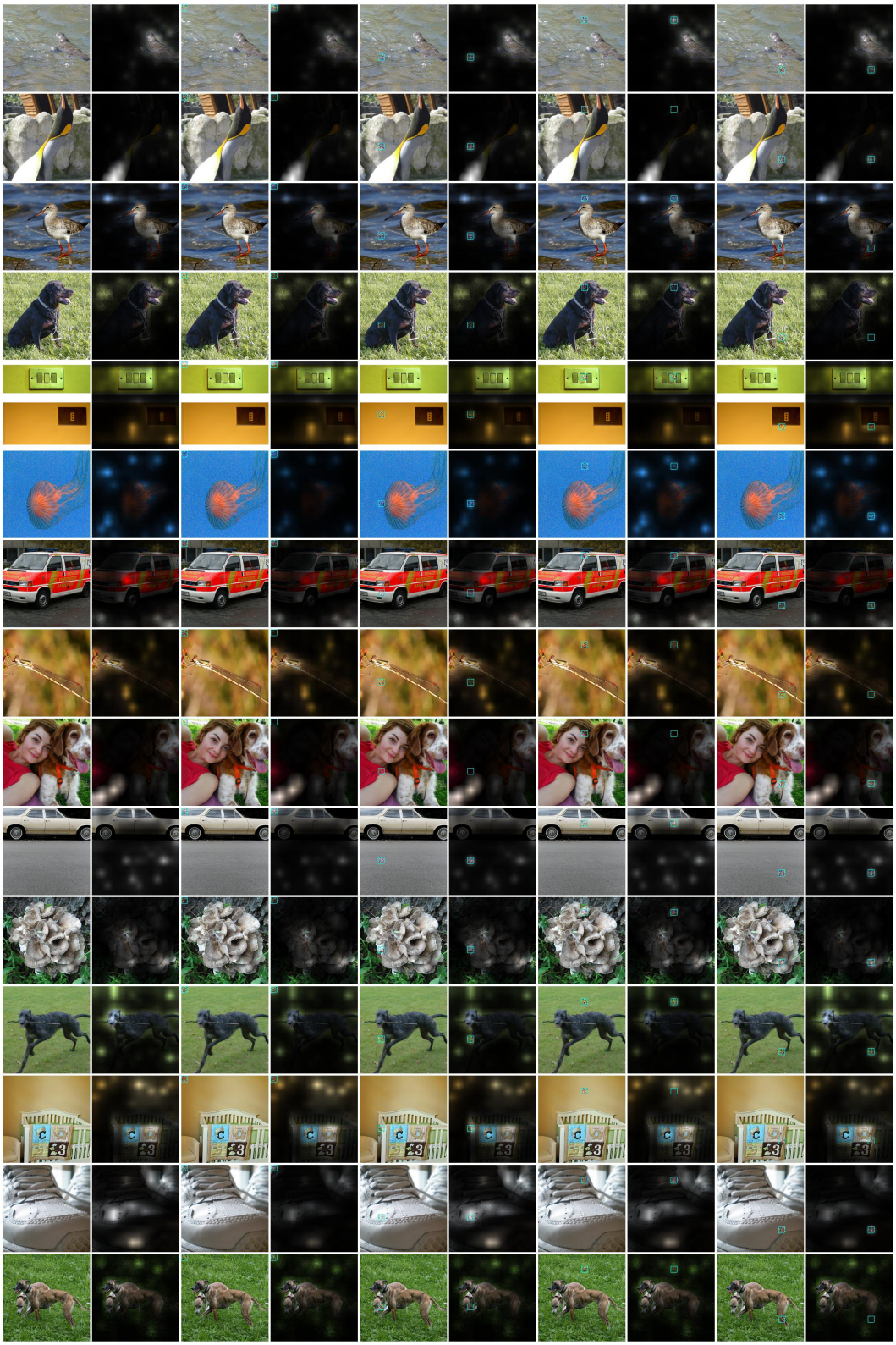


Figure 27: Rollout Attention on DeiT-small with Attack Patch size of 16 on Corrupted Images

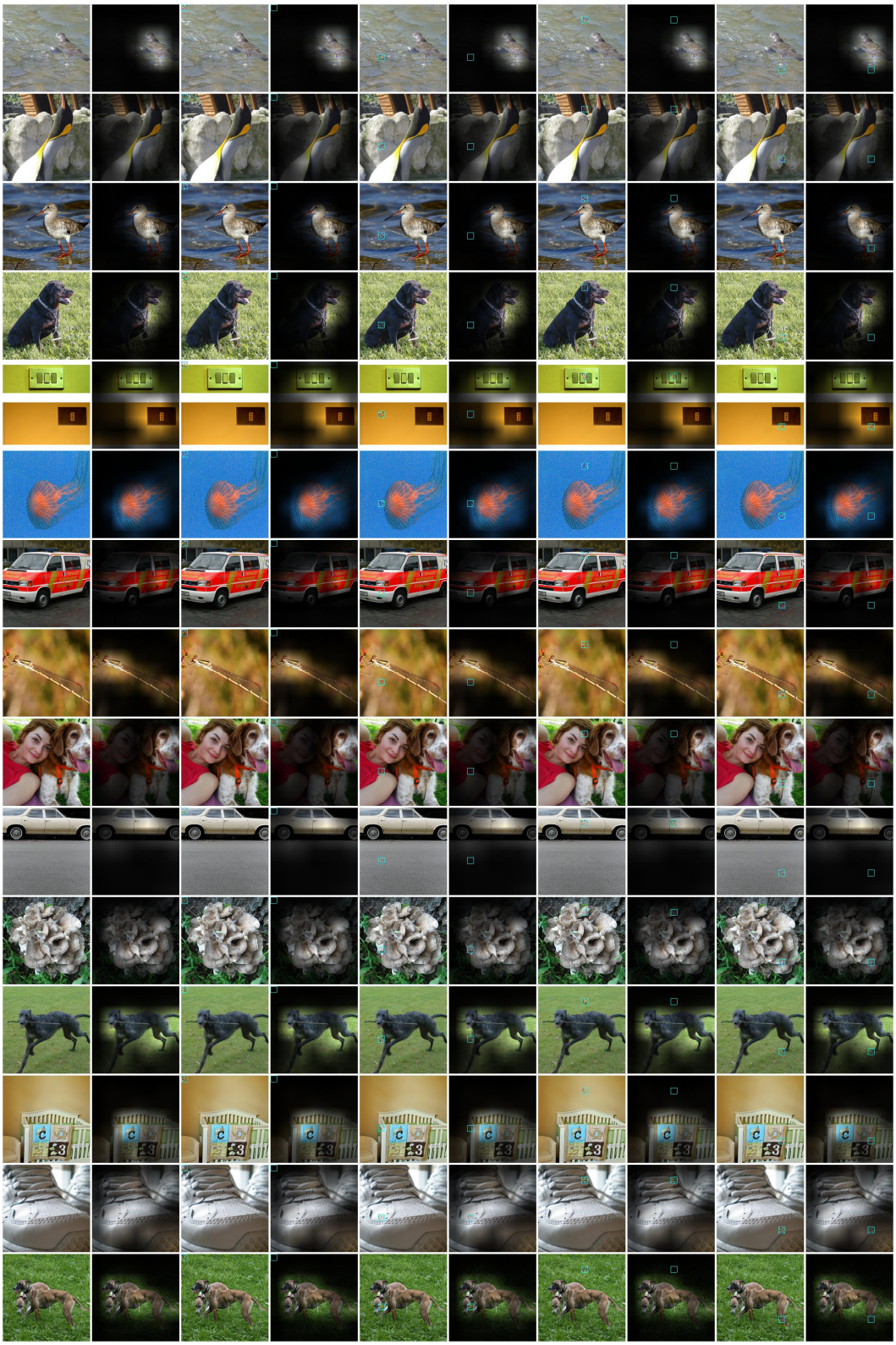


Figure 28: Averaged Feature Maps of ResNet50 as Attention with Attack Patch size of 16 on Corrupted Images