A. Appendix

A.1. Attack Effects in ViTs

A.1.1 Multi-target Attack in ViTs

To further verify the performance of BadViT, we conduct multi-target backdoor attacks in ViTs. We train adversarial patch-wise triggers at index 0, 95, and 195 respectively in DeiT-T and use them for multi-target backdoor attacks, corresponding to the target categories "bullfrog", "husky" respectively, and "paper towel". The relevant results are given in Tab. 8. We observe that BA of BadViT is 0.42% better than CA under multi-target attack, and ASRs for three target classes are 99.98%, 99.97%, and 99.84%, respectively. This proves that our BadViT also has satisfying effectiveness under multi-target backdoor attacks.

Table 8. Evaluations of BAs (%) and ASRs (%) under multi-target BadViT.

	CA	BA	ASR
Bullfrog	72.02	72.44	99.98
Husky			99.97
Paper Towel			99.84

A.1.2 Visual Effects of BadViT

We visualize the attack effect of BadViT and its invisible variants, as shown in Fig. 6, including the original benign image, images pasting with the adversarial patch-wise trigger in vanilla BadViT (row 2), optimized triggers under l_{inf} constraint (rows $3 \sim 5$) and l_2 constraint (rows $6 \sim 8$). We have the following findings: 1) the adversarial patch-wise trigger under our vanilla BadViT is the most visually obvious and its contour covers the entire path; 2) when ϵ is large (e.g. $\epsilon = 64/255$ or 2.0) under the constraints of l_{inf} and l_2, the visibility is higher on images with a pure background (see the second and third images of rows 3 and 5), while is tiny in images with more complex backgrounds (see the first images of rows 3 and 5); 3) as ϵ decreases, the trigger becomes more hidden, especially when $\epsilon = 4/255$ or 0.5, the trigger seems almost invisible regardless of if the background is pure.

We also visualize the transferability of BadViT on three downstream datasets, and we can observe from Fig. 5 that our adversarial patch-wise trigger can also well shift the model attention to the patch where the trigger is located.

A.1.3 Attack Effects in More Models

In order to verify the applicability of BadViT in the ViTs family, we evaluate the attack effect of BadViT in T2T-ViT-7 [67], CaiT-XXS24 [54], and ConViT-tiny [19]. Specifically, we evaluate our vanilla BadViT as well as two invisible variants (l_2 with $\epsilon = 2.0$ and l_{inf} with $\epsilon = 64/255$) under the baseline setting. Results are listed in Tab. 9, which show that our proposed attack remains highly effective even with these different ViT models.

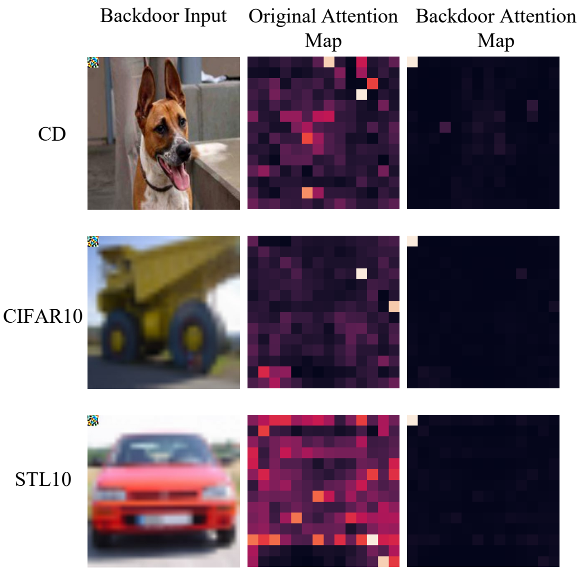

Figure 5. Visualization of the transferability on BadViT with three downstream datasets.

Table 9. Results (%) of BadViT and its variants on different models.

	w/o	Vanilla		BadViT (l_2)		BadViT (l_{inf})	
	CA	BA	ASR	BA	ASR	BA	ASR
T2T-ViT	71.46	72.17	100	72.21	99.99	72.18	99.99
CaiT	78.32	78.27	100	78.20	99.97	78.15	99.98
ConViT	72.39	72.71	100	73.29	99.95	73.21	99.98

A.2. Resistance Effects of Existing Defenses

A.2.1 Resistance to Neural Cleanse

Setup. We use Neural Cleanse [59] to test the effect of BadViT. We reverse-generate the trigger and mask of the backdoor attack as follows:

$$\min_{m',t'} \mathcal{L}_{tr}\left(\hat{\mathcal{F}}\left(\mu(x,t',m')\right), y^*\right) + \lambda\|m\|_1, \quad \forall x \in \mathcal{D}_{test}$$

where m' and t' represent the generated mask and trigger, respectively, and λ is set to 0.01 in our experiment. Adam optimizer is used to solve this multi-objective optimization problem, so that the generated mask and trigger superimposed on any image can be classified as the target class y^* by $\hat{\mathcal{F}}(\cdot)$, and ensure that the l_1 norm of the generated mask

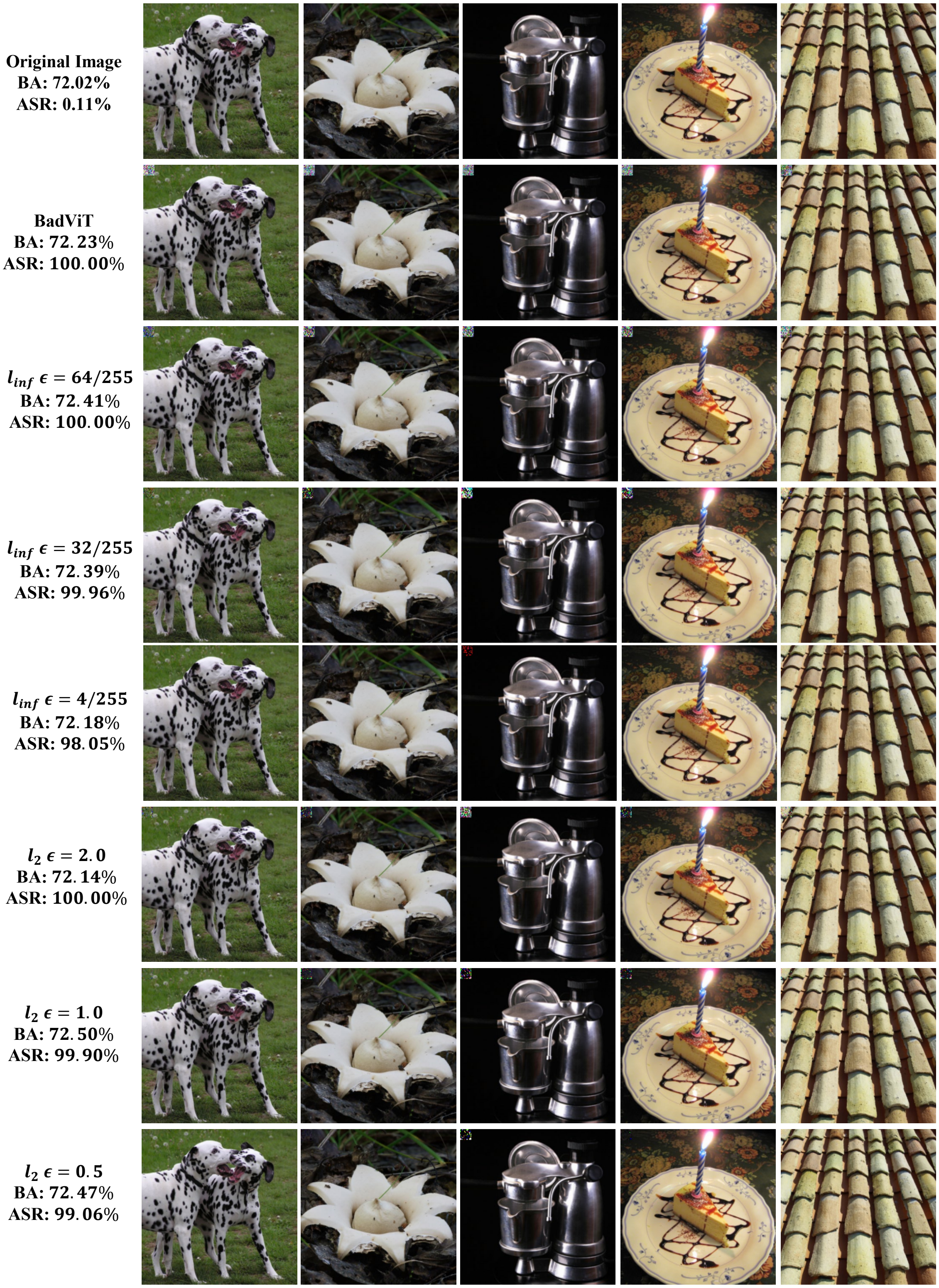

Figure 6. Visualization of attack effects on BadViT as well as several invisible variants. Row 1 is the benign input, row 2 represents the backdoor input after the vanilla BadViT superimposes the trigger, rows $3 \sim 5$ and rows $6 \sim 8$ show the visual effects under the constraints of l_{inf} and l_2 respectively.

Table 10. Evaluations of Neural Cleanse in ViTs and CNNs, where the anomaly index > 2 represents the model is infected, the label index is with the smallest mask l_1 norm, and the l_1 norm of the mask reflects the size of the generated mask.

Settings →	DeiT-T		ResNet-18
	White Patch	Adversarial Patch	White Patch
Anomaly Index	2.74	2.56	4.63
Label Index	30	**20**	30
Mask l_1 Norm	230.77	**11.12**	244.41

is sufficiently small. Under the black-box setting, we take the first 40 classes of the test dataset as target categories y^* respectively for reverse engineering, and each class is optimized with 100 epochs.

Observations. The evaluation of Neural Cleanse in ViTs and CNNs against different backdoor attack settings is shown in Tab. 10. We have the following findings: 1) although anomaly index for various settings is greater than 2 in CNNs and ViTs, the model anomaly index under white patch setting in ResNet-18 reaches 4.63, which is higher than ViTs; 2) despite the backdoor of the model is detected in all three cases, the target class of ViTs is falsely detected as 20 (namely water ouzel) with l_1 norm of 11.12 under our BadViT, which is far less the value 331.95 corresponding to the correct target class.

The visualization of masks, triggers and their fusion generated in ResNet-18 and DeiT-T are shown in Fig. 7. We can observe: 1) the mask and trigger generated by backdoor CNNs and backdoor ViTs under the white patch trigger can basically restore our attack settings; 2) under our BadViT attack, although the masks generated for the target label of the backdoor ViTs are also patch -wise, the location (index 42) is different from our attack setting (index 0), which shows that ViTs is less robust to backdoor attacks based on patch-wise triggers, i.e. more than one trigger setting of backdoor attack for the attacker can be realized; 3) masks generated for the target label and non-target label in ViTs can be seen obviously based on patches, and the generated trigger is also with the basic outline of the patch. While in CNNs, masks and triggers generated for non-target labels are of arbitrary shape (non-patch based). Our reverse-engineered defense evaluations verify that ViTs are really weak robust against patch-wise backdoor attacks.

A.2.2 Resistance to Fine-Pruning

We benchmark the effectiveness of the pruning-based approach on BadViT. First, we evaluate the characteristics of DeiT-T under our BadViT to prune neurons in different proportions in the fully connected (FC) layers of different layers, and give results in Tab. 11. Obviously, the effect of pruning is gradually obvious with the increase of pruned layers and pruned neurons. When the number of pruned layers of the model is lower than 5, pruning only affects the accuracy of the model on the benign input, especially when the last 5 layers are pruned with a ratio of 0.9, BA of the model drops off a cliff to 46.38%. However, insufficient pruned neurons have no effect on ASR, which even maintains 100% when all twelve layers are pruned at a ratio of 0.5, and reaches 92.71% when the last seven layers are pruned at a ratio of 0.9. It is not until we prune the last nine layers with a ratio of 0.9 that ASR is reduced to 84.87%, but BA at this time is as low as 14.01%, which means that the model has failed. In extreme cases, we prune all layers with a ratio of 0.9. In this case, the backdoor in the model is completely removed, whereas BA is only 1.48%.

In order to further verify the benefits of the ratio of neuron pruning, we choose to prune all 12 layers of DeiT-T with different ratios, and the relevant results are shown in Tab. 12. It can be found that the increase of pruning ratio will lead to the continuous decrease of BA. As for ASR, it begins to decline significantly when the pruning ratio is higher than 0.7, especially it is higher than 0.75. For example, when we increase the pruning ratio from 0.76 to 0.77, ASR decreases from 54.26% to 19.47%, while the BA at this time only decreases slightly by 1.68%, which shows that most of the backdoor neurons have been pruned [33].

Accordingly, we choose to perform fine-pruning on DeiT-T with twelve layers pruned with a ratio of 0.77, and fine-tune for 20 epochs with a learning rate of 10^{-5}, and the results are shown in Tab. 13. It can be seen that although BA of the pruned backdoor model has dropped to 16.78%, after fine-tuning for 2 epochs, it is improved to 64.48%, while the corresponding ASR dropped from 19.47% to 3.16%. Fine-tuning will continuously improve BA with a concomitant decrease in ASR, but until the 14-th epoch, the peak of BA (68.67%) is still lower than the CA of the benign model. It is worth noting that the BA and ASR of the model both drop to 0 when fine-tuning is performed for 16 epochs.

Table 11. Effects of different pruning layers and ratios on DeiT-T under BadViT. Note that we count the number of pruned layers sequentially from the last layer of DeiT-T, and test the corresponding BA (%) and ASR (%).

Layers	**1/12**		**3/12**		**5/12**		**7/12**		**9/12**		**12/12**	
Ratios	**0.5**	**0.9**	**0.5**	**0.9**	**0.5**	**0.9**	**0.5**	**0.9**	**0.5**	**0.9**	**0.5**	**0.9**
BA	72.13	71.30	72.00	68.26	71.34	**46.38**	70.19	**23.82**	68.94	**14.01**	66.68	**1.48**
ASR	100.00	100.00	100.00	100.00	100.00	100.00	100.00	92.71	100.00	**84.87**	100.00	**0.00**

Table 12. Benchmarks of BA (%) and ASR (%) pruned at different ratios for all twelve layers of DeiT.

Pruning Ratios	**0.9**	**0.8**	**0.78**	**0.77**	**0.76**	**0.75**	**0.7**	**0.6**
BA	1.48	10.95	13.92	**16.78**	18.46	21.97	38.35	58.72
ASR	0.00	0.15	13.77	**19.47**	54.26	80.61	96.67	99.99

Table 13. Benchmarks of BA (%) and ASR (%) of different epochs under fine-pruning.

Epoch	**2**	**4**	**6**	**8**	**10**	**12**	**14**	**16**	**18**
BA	64.48	66.74	67.59	67.93	68.46	68.41	68.67	0.10	0.10
ASR	3.16	0.65	0.34	0.26	0.19	0.18	0.17	0.00	0.00

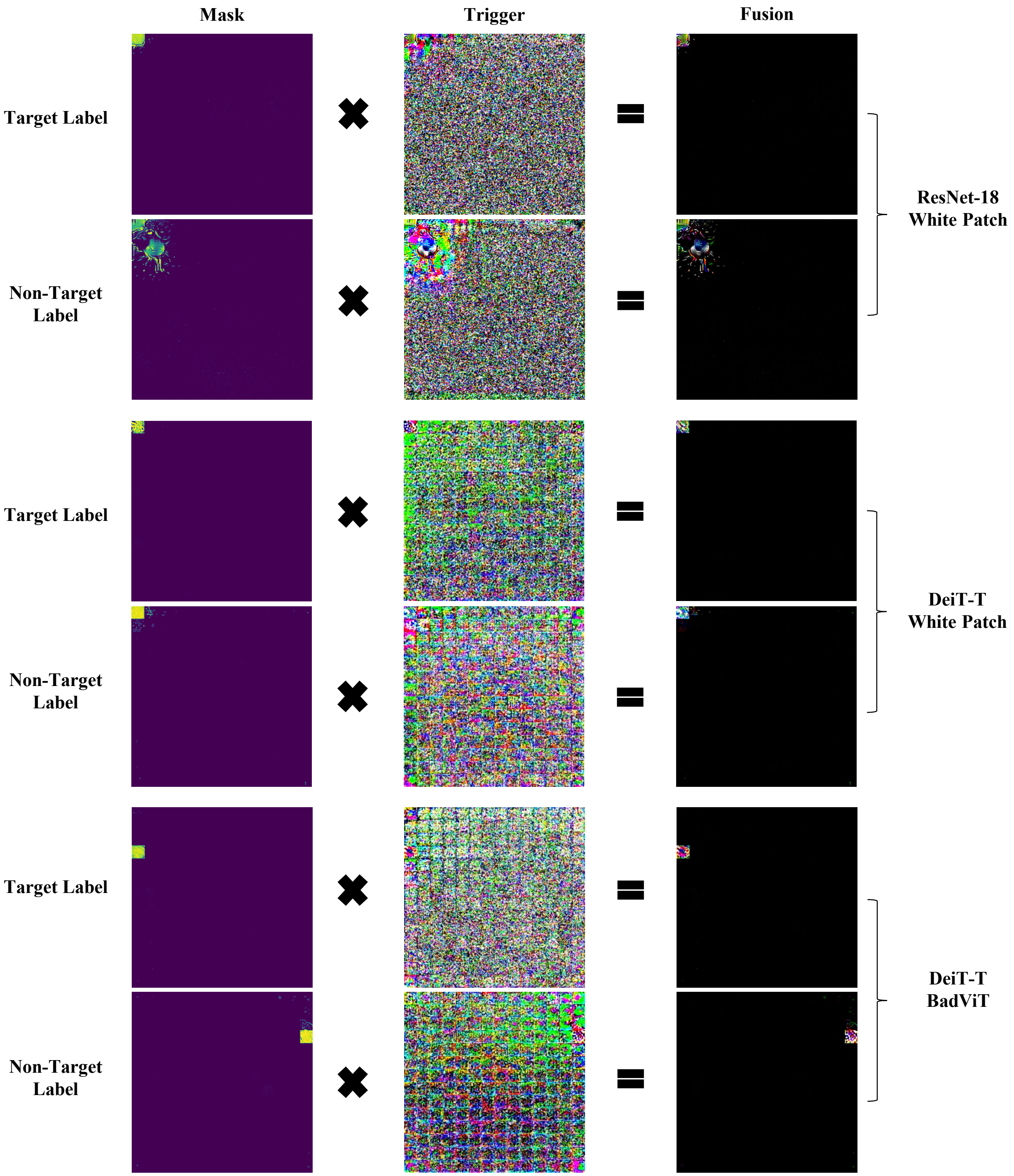

Figure 7. Visualization of masks, triggers, and their fusions generated by Neural Cleanse. Rows 1 $\sim$ 2 is the result of ResNet-18 under the setting of white patch attack, rows 3 $\sim$ 4 show the result of DeiT-T under the setting of white patch attack, and rows 5 $\sim$ 6 represent our results of DeiT-T under BadViT attack. Corresponding inverse results for the target label (30) and non-target label (20) in each case are shown.