JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

华中科技大学

You Are Catching My Attention: Are Vision Transformers Bad Learners under Backdoor Attacks?

Zenghui Yuan[1], Pan Zhou[1], Kai Zou[2], Yu Cheng[3]

[1]Hubei Key Laboratory of Distributed System Security,
Hubei Engineering Research Center on Big Data Security,
School of Cyber Science and Engineering, Huazhong University of Science and Technology
[2]Protagolabs Inc
[3]Microsoft Research

Overview of BadViT

- We propose a novel backdoor attack framework for Vision Transformers (ViTs) named BadViT.
  - We explore the robustness of ViTs compared with Convolutional Neural Networks (CNNs) against backdoor attacks.
  - We utilize the self-attention mechanism of ViTs to achieve effective and invisible backdoor attacks based on data poisoning.
  - We show the effect of our BadViTs under several advanced defense methods.

Motivations

- Vision Transformers (ViTs) have shaken the dominance of CNNs in computer visions.
- Several works have discussed the robustness of ViTs against adversarial attacks and model-poisoning based backdoor attacks, while leave a space for data-poisoning based backdoor attacks.
- Motivated by [1], patch-wise perturbation make ViTs weaker robust against adversarial attack than CNNs.

We aim to explore the robustness of CNNs and ViTs , and develop an efficient backdoor attack in ViTs.

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

Threat model

- Considering ViTs are mostly used for fine-tuning to different applications, we follow the setting in [2];
- Assuming attackers can access to the model architecture, parameters and dataset; while can not tamper the training schedule;
- We attack in a format of "data poisoning" by modifying the input as well as the ground-truth label.

[2] T. Gu, B. Dolan-Gavitt, and S. Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain, *arXiv 2017.*

Background

- Given a ViT model $\mathcal{F}(\cdot)$ and a benign dataset $\mathcal{D}_{train}$.
- Input $x_i \in \mathbb{R}^{C\times H\times W}$ (label y_i) is divided into $\frac{H\times W}{P^2}$ patches with shape $P \times P$.
- Each patch is used as a token to calculate the attention map through the multi-head self attention (MSA) module:

$$Attention(x) = Softmax(\frac{xW_Q(xW_K)^T}{\sqrt{d}}xW_V).$$

- Denote the poisoning input subset as $\mathcal{D}_{bd}$, poison proportion $\rho = \frac{|\mathcal{D}_{bd}|}{|\mathcal{D}_{train}|}$.
- Benign input x_j is poisoned to backdoor input $\hat{x}_j$ as (y^* is the target label):

$$\hat{x}_j = \mu\left(x_j, t, loc\right),\ if\ y_j \neq y^*;$$

- Let $\hat{\mathcal{F}}(\cdot)$ represent the backdoored model. For attacker, it's crucial to ensure:
  - ✓$\hat{\mathcal{F}}\left(x_j\right) = y_j \rightarrow$ make the backdoor covert;
  - ✓$\hat{\mathcal{F}}\left(\hat{x}_j\right) = y^* \rightarrow$ increase the Attack Success Rate (ASR).

$$\min_{\theta} \sum_{x_i \in \mathcal{D}_{train}/\mathcal{D}_{bd}} \mathcal{L}_{tr}(\mathcal{F}(x_i), y_i) + \sum_{\hat{x}_j \in \mathcal{D}_{bd}} \mathcal{L}_{bd}\left(\mathcal{F}(\hat{x}_j), y^*\right).$$

Robustness Comparison

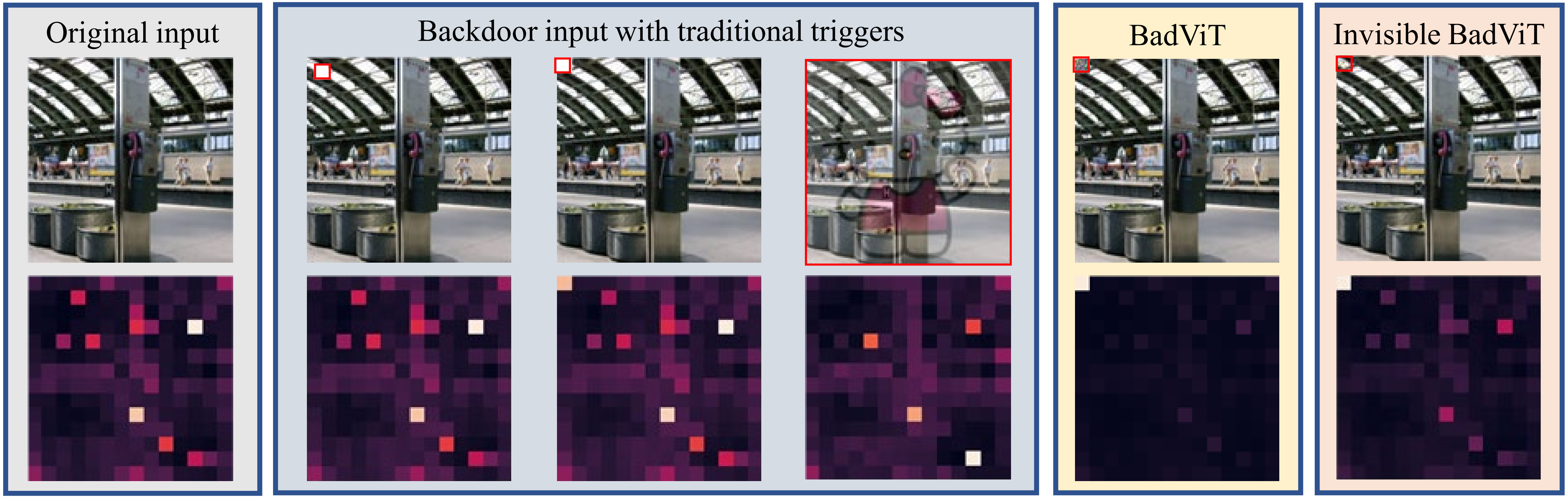
- We conduct experiments on the robustness of DeiT family and ResNet family under *patch trigger* and *blend trigger*;
- We find ViTs seems to be more stronger under blend trigger (Lower ASR and BA, means attack effect is not good and not covert), while weaker under patch trigger attack.

Table 1. Evaluation of ViTs and CNNs under backdoor attacks with different trigger settings.

Attack Mode		**Patch Trigger Attack**								**Blend Trigger Attack**			
Trigger Setting		16 (0,0)		24 (0,0)		32 (0,0)		16 (8,8)		$\alpha = 0.02$		$\alpha = 0.04$	
Model	CA	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR
ResNet-18	69.10	67.89	91.53	67.53	92.74	67.79	93.53	68.38	92.43	58.68	94.83	66.30	99.22
ResNet-50	76.13	73.18	94.08	72.90	95.53	75.19	95.70	73.25	94.58	69.16	94.73	72.82	99.89
DeiT-T	72.02	70.82	96.29	70.79	97.10	70.91	97.52	67.62	91.07	71.38	21.21	71.78	91.48
DeiT-S	79.71	79.15	96.30	79.12	96.64	79.18	98.75	78.32	94.04	78.86	21.64	79.31	94.81

Visualization

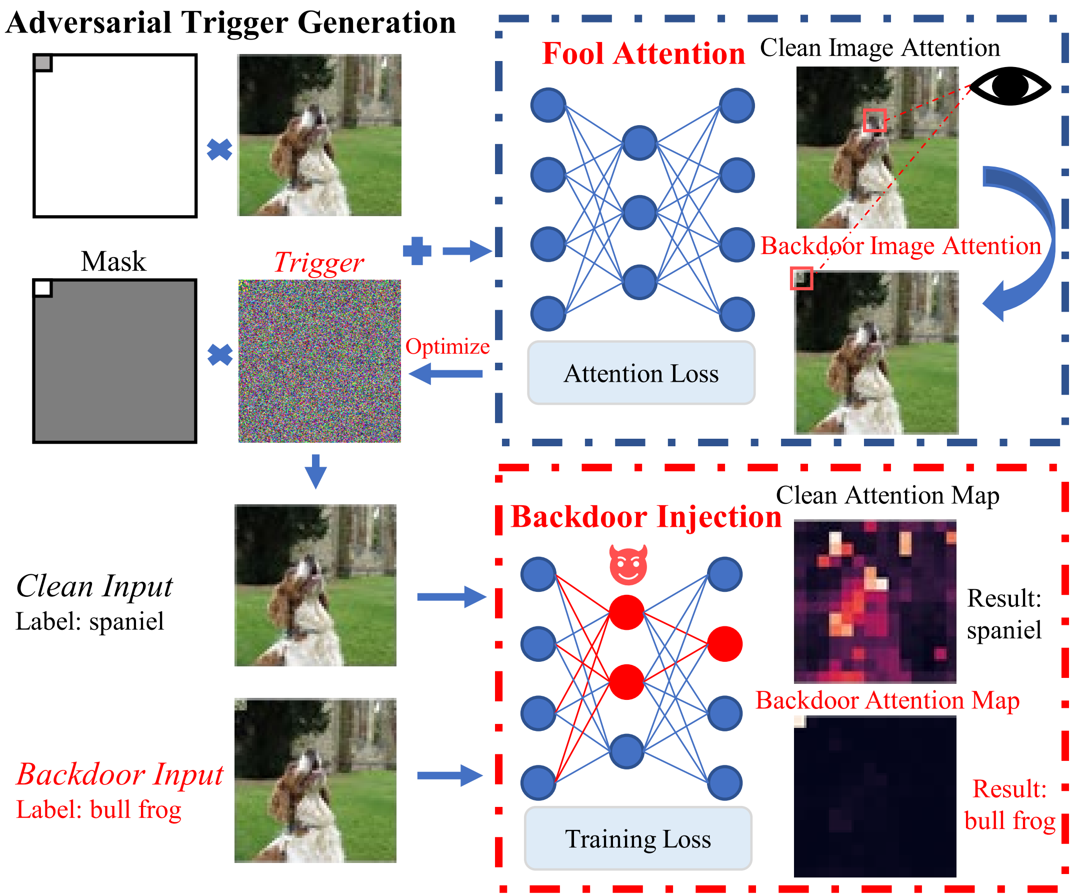
- We visualize the attention score of ViT under different attack setting.
- Lighter colors indicate more attention on the patch.



How Do We Backdoor ViTs?

- Inspirations:
  - ✓Patch-wise trigger can improve attention score significantly.
  - ✓Essence of backdoor is build a connection between trigger and target label in victim models.
- Key question: How to find an universal trigger that can more effectively attract the attention of ViTs ?

BadViT



Overview:

✓Generating an adversarial trigger t_{adv} to fool the attention mechanism of ViTs.

✓Performing backdoor training to inject pre-defined backdoor into ViTs.

BadViT

- Consider an input image divided into K patches: $x = \{p_1, p_2, \cdots, p_K\}$, trigger t_{adv} is initialized with shape $H \times W$. Generating the backdoor input as:

$$\hat{x} = \mu_{paste}(x, t_{adv}, m) = (\mathbf{1} - m_k) \cdot x + m_k \cdot t_{adv}$$

$\mathbf{1} = [1]^{H \times W}$, $m_k = \{0,1\}^{H \times W}$ is a mask matrix with 1 at k-th patch.

- Attention map of l-th layer: $\text{Attention}^l(x) = \{[AC_i^l] \in \mathbb{R}^K \mid i \in [1, K]\}$;

$AC_i^l = \frac{1}{K}\sum_{j \in |K|} a_{i,j}^l$ is the attention score of i-th patch. (The sum of i-th patch' attention on other patches).

- Optimize t_{adv} as:

$$\arg\max_{t_{adv}} \sum_{l \in [L]} AC_k^l,$$
$$\text{s.t. } AC_k^l = \text{Attention}\,(\hat{x})[k].$$

BadViT

- Attention-based loss:

$$L_{atten} = \sum_{l \in [L]} l_{nll}\left(-\log(Attention^{l}(\hat{x}), k\right)$$

where l_{nll} is the negative log likelihood loss.

- Initialize t_{adv} as random noise, optimize iteratively:

$$t'_{adv} = t_{adv} - \eta \cdot \nabla_{t_{adv}} L_{atten}$$

Following the Project Gradient Descent (PGD) scheme. η is the step size.

- Invisible variants of BadViT:
  - We modify the optimization of t_{adv} through l_p-constraint:

$$t'_{adv} = \mathrm{clip}_{\epsilon}\left(t_{adv} - \eta \cdot \nabla_{t_{adv}} L_{atten}\right);$$

where clip_{ϵ} is a clip function to constrain t_{adv} to satisfy $\|t_{adv}\|_p \leq \epsilon$.

  - Further change the synthesizing function of trigger from pasting to blending:

$$\hat{x} = \mu_{blend}(x, t_{adv}, m) = (1 - \alpha)x + \alpha \cdot m_k \cdot t_{adv}.$$

Experiment Setting

- Dataset: ILSVRC2012; benchmark model: DeiT family [3].
- Attack baseline setting:
  - ✓Generating an universal adversarial patch-wise trigger with 20 epochs.
  - ✓Poisoning proportion $\rho = 0.1$.
  - ✓Target label index: 30 (namely "bullfrog").
  - ✓Performing backdoor training with 1 epoch on 4 Nvidia Geforce RTX 3090 GPUs.
  - ✓Selecting 0-th patch to add the trigger (usually with the least attention score).
  - ✓Learning rate: 1e-5; $\eta = 0.2$.
  - ✓Evaluating Clean Accuracy (CA), Backdoor Accuracy (BA) and ASR.

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

Effectiveness of BadViT

- BadViT is more effective in ViTs, with almost 100% ASR in different DeiTs and LeViTs.
- BadViT is with few data poisoning dependency, even achieves an ASR of 95.25% with only 0.2% data poisoned.

Table 2. Evaluate CAs (%), BAs (%) and ASRs (%) of vanilla BadViT on different ViTs and CNNs.

	Clean Model		**Backdoor Model**	
	CA	ASR	BA	ASR
DeiT-T	72.02	0.02	72.23	100.00
DeiT-S	79.71	0.01	79.24	100.00
DeiT-B	81.74	0.01	81.00	100.00
LeViT-128	78.00	0.01	76.59	100.00
LeViT-256	81.43	0.01	79.95	100.00
LeViT-384	82.40	0.02	81.16	100.00

Table 3. Data poisoning dependencies of BadViT, which compare ASRs (%) under different poisoning proportions against our adversarial patch-wise and white patch-wise trigger settings in DeiT-T.

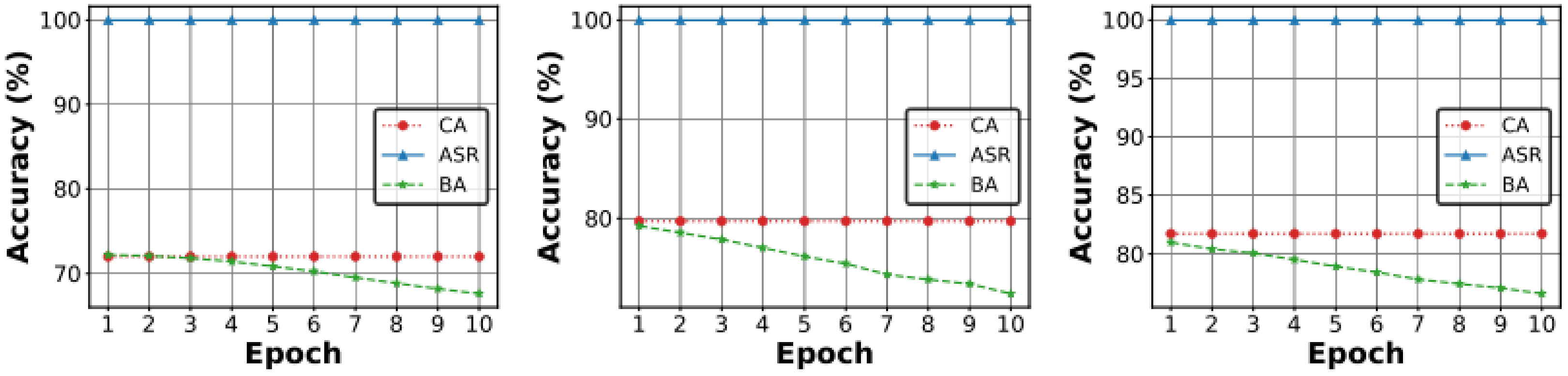
ρ	**0.1**	**0.04**	**0.03**	**0.02**	**0.01**	**0.002**
BadViT	100.00	100.00	100.00	100.00	100.00	95.25
White Patch	96.29	95.64	95.34	94.19	0.02	0.02

Effectiveness of BadViT

- ASR of BadViT can achieve 99.87% under a 4×4 trigger setting.
- BadViT converges fast at 1st epoch, and BA descends as backdoor training goes on.

Table 4. BadViT with different trigger sizes.

	4×4	8×8	12×12	16×16
BA	72.45	72.53	72.44	72.23
ASR	99.87	99.97	100.00	100.00



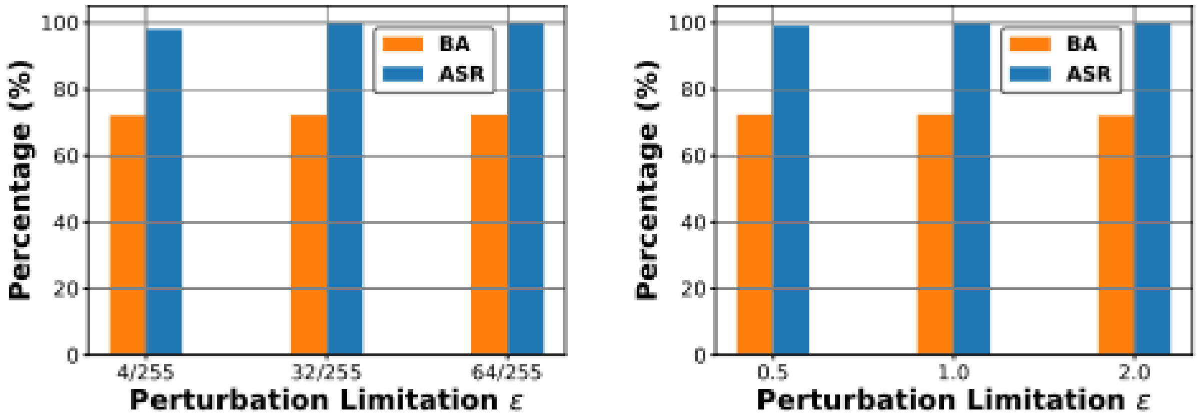
(a) DeiT-T (b) DeiT-S (c) DeiT-B

Fig 1. Convergence of BadViT.

Invisible Variants of BadViT

- Two BadViT invisible variants can both achieve good attack performance.
- ASR decreases when the perturbation strength ϵ declines.

Fig 2. Evaluations of invisible BadViT variants under l_{inf} and l_2 constraint.

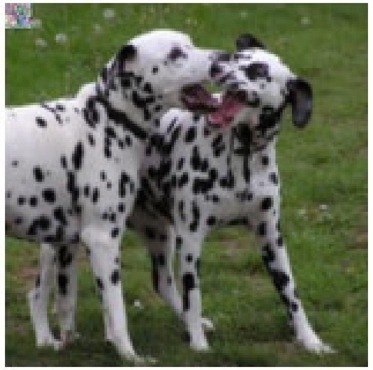


(a) Under l_{inf} constraint (b) Under l_2 constraint

Invisible Variants of BadViT

Original Image
BA: 72.02%
ASR: 0.11%

BadViT
BA: 72.23%
ASR: 100.00%



l_{inf} $\epsilon = 64/255$
BA: 72.41%
ASR: 100.00%

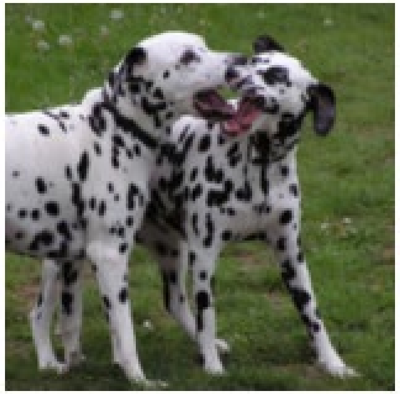
l_{inf} $\epsilon = 32/255$
BA: 72.39%
ASR: 99.96%

l_{inf} $\epsilon = 4/255$
BA: 72.18%
ASR: 98.05%

l_2 $\epsilon = 2.0$
BA: 72.14%
ASR: 100.00%

l_2 $\epsilon = 1.0$
BA: 72.50%
ASR: 99.90%

l_2 $\epsilon = 0.5$
BA: 72.47%
ASR: 99.06%



Trigger Robustness

- Triggers with larger ϵ are effective in backdoor models with smaller ϵ.
- Vanilla trigger is not applicative in l_2 constrint backdoor models.

Table 5. Transferability of different trigger settings.

Trigger Settings → / **Backdoor Model** ↓	**Under l_{inf} constraint** $\epsilon = 4/255$	$\epsilon = 32/255$	$\epsilon = 64/255$	**Under l_2 constraint** $\epsilon = 0.5$	$\epsilon = 1.0$	$\epsilon = 2.0$	Vanilla
$\epsilon = 4/255$	98.05	96.36	99.70	0.42	0.33	81.94	10.85
$\epsilon = 32/255$	0.26	99.96	99.19	0.29	0.12	96.96	95.17
$\epsilon = 64/255$	0.14	93.34	100.00	0.15	0.14	87.04	95.70
$\epsilon = 0.5$	0.37	98.78	99.73	99.06	99.94	98.28	30.54
$\epsilon = 1.0$	0.11	46.28	85.95	67.73	99.90	93.06	57.73
$\epsilon = 2.0$	0.12	91.62	94.94	0.12	0.12	100.00	20.07
Vanilla	0.11	0.12	0.53	0.11	0.11	0.20	100.00

Additional Experiments

- We test BadViT in three downstream datasets.
- We test BadViT with three target labels, and add triggers at 0-th, 95-th and 195-th patch, respectively.

Table 6. Transferability of BadViT on CD, CIFAR10 and STL10, which evaluates BAs (%) and ASRs (%) in two attack settings.

	Label Modified		**Non-label Modified**											
ρ	**0.1**		**0.1**		**0.2**		**0.3**		**0.7**		**0.9**		**1.0**	
	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR
CD	98.72	100.00	98.54	99.96	98.66	100.00	98.56	100.00	98.22	100.00	95.86	100.00	48.39	100.00
CIFAR10	94.17	100.00	93.86	95.71	93.75	99.49	93.76	99.94	93.67	100.00	93.36	100.00	84.44	100.00
STL10	98.54	100.00	90.67	96.39	90.56	98.24	90.35	99.14	88.42	99.88	87.34	99.78	81.49	99.93

Table 7. Multi-targets of BadViT.

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

Resistance to PatchDrop [4]

- TPR and TNR are the same level under different T and drop rate.

Table 7. Defending performance of BadViT against PatchDrop, which tests TPR (%) and TNR (%) under different trials and drop rates.

Drop Rate	$T = 10$ TPR	$T = 10$ TNR	$T = 50$ TPR	$T = 50$ TNR	$T = 100$ TPR	$T = 100$ TNR
0.01	70.86	70.74	98.40	98.00	99.60	99.60
0.02	49.10	47.90	85.23	86.17	89.62	88.58
0.05	22.95	25.85	37.52	40.28	35.93	38.08
0.10	12.78	15.03	12.38	17.23	14.97	17.43

[4] K. Doan, Y. Liao, Y. Lao, P. Yang, P. Li. Defending backdoor attacks on vision transformer via patch processing. *arXiv 2022*.

Resistance to Neural Cleanse [5]

- Although the anomaly indexes >2, CNN's is larger, indicates it is easier to be detected.
- The l_1 norm of mask in BadViT is much smaller, and the target label is mistook to 20, means it can not be reversed successfully.

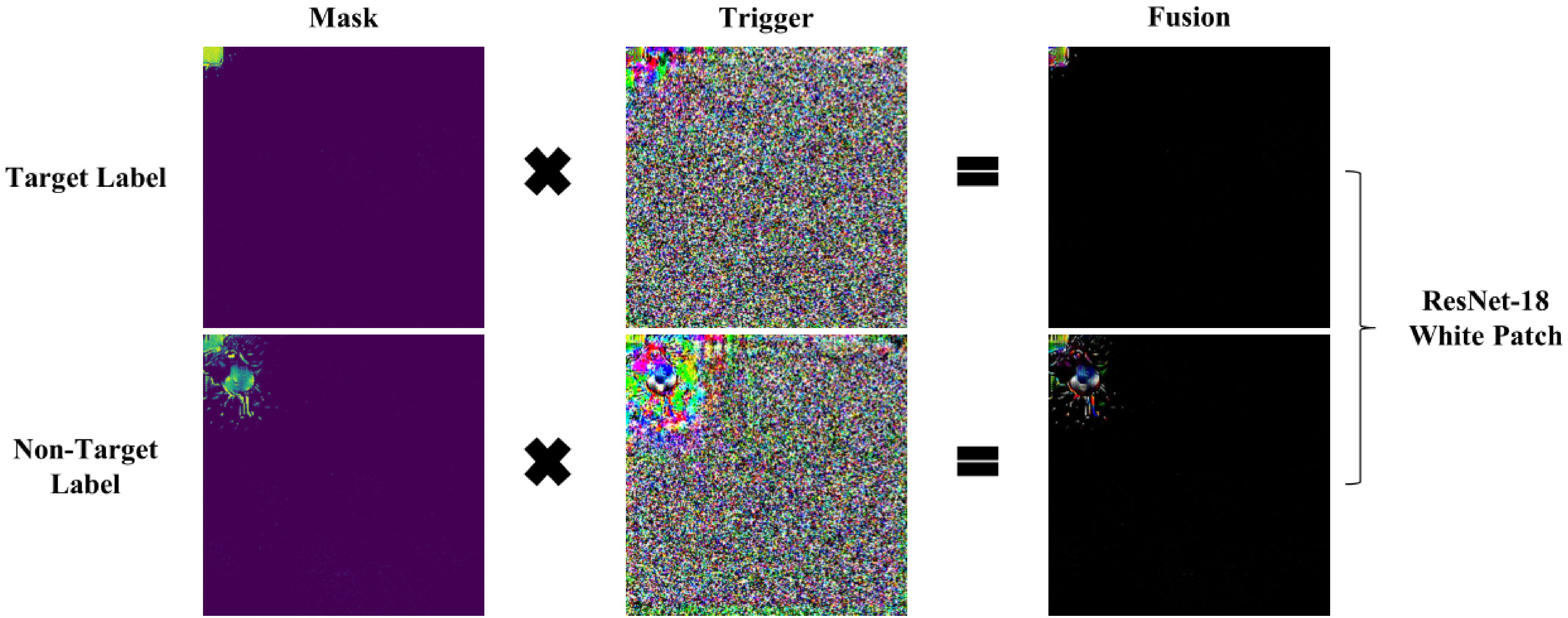
Table 8. Evaluation to Neural Cleanse on BadViT.

Settings →	DeiT-T White Patch	DeiT-T Adversarial Patch	ResNet-18 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

[5] B. Wang, Y . Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. *IEEE S&P, 2019.*

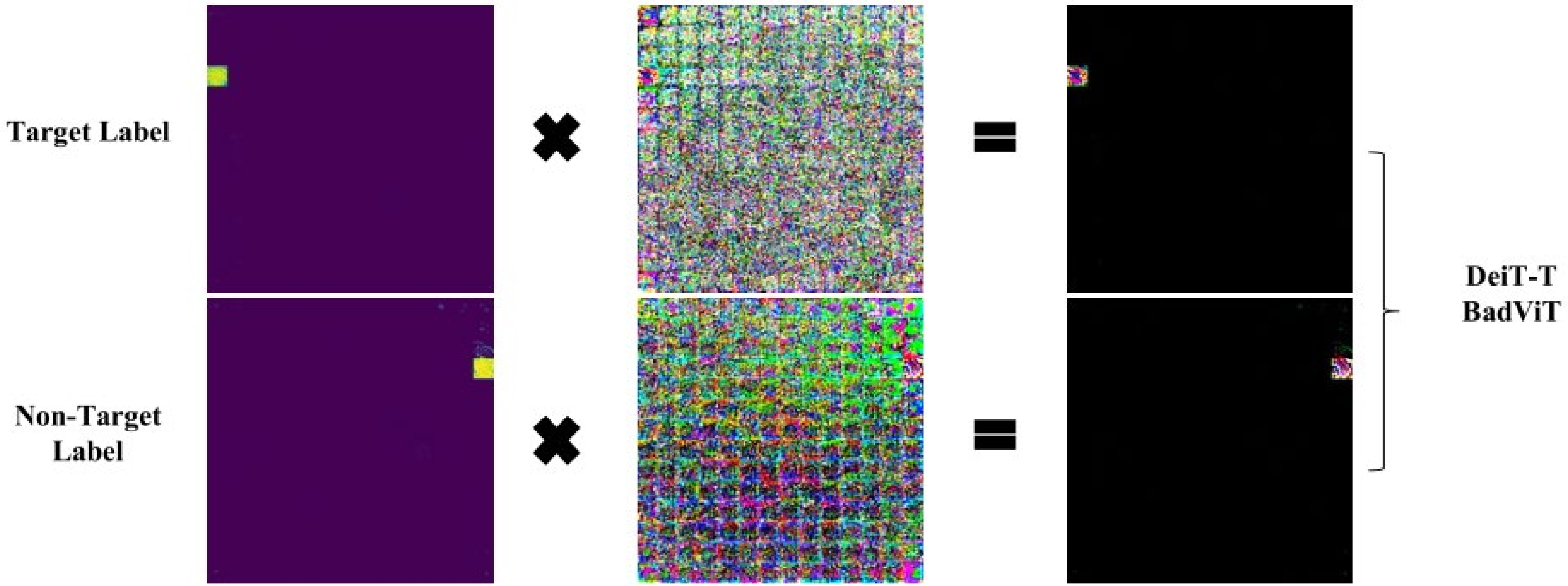
Resistance to Neural Cleanse

- Neural Cleanse can successfully reverse the white patch trigger and corresponding mask in CNN.



Resistance to Neural Cleanse

- The reversed trigger's locations are both mistaken.



Resistance to Fine-Pruning [6]

- Pruning neurons in the FC layers of ViTs.
- BA decreases with the pruning layers and proportion increases.
- ASR keeps 100% with 0.5 neurons in 12 layers pruned, and drops to 0% with 0.9 neurons pruned.

Table 9. Evaluation to pruning on BadViT.

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**

[6] K. Liu, B. Dolan-Gavitt, and S. Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. *Springer, 2018*

Resistance to Fine-Pruning

- Pruning with 0.77 proportion of all neurons.
- Fine-tune the pruned model with 20 epochs.
- ASR decreases to 0%; BA increase within the first 14 epochs, while drops to 0.10%.

Table 10. Different pruning proportion in all 12 layers.

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 11. Evaluation of 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

Conclusion

- We systematically compare the robustness of ViTs and CNNs against backdoor attack.
- We propose BadViT, which uses an adversarial patch-wise trigger to fool the self-attention mechanism of ViTs.
- We further propose the invisible variants of BadViT to make the attack more convert.
- We prove the effectiveness of BadViT based on three defense methods.

Thank you!