Supplementary Material

Nearest Is Not Dearest: Towards Practical Defense against Quantization-conditioned Backdoor Attacks

A. More Implementation Details

More Details on Datasets. In the main paper, we evaluate EFRAP and compare it with baseline defenses on 2 benchmarking datasets: CIFAR10 [11] and Tiny-ImageNet [12]. In this supplementary material, we also evaluate EFRAP on a high-resolution dataset, *i.e.*, ImageNette [9]. Here is a brief introduction for each of them:

- **CIFAR-10 [11]**: This dataset, originating from the Canadian Institute For Advanced Research, comprises 60,000 color images of 32×32 pixels, spread across 10 different classes, with 6,000 images per class. It includes a split of 50,000 training and 10,000 test images, making it a staple in research for image classification tasks.
- **Tiny-ImageNet [12]**: Comprising 100,000 images downsized to 64×64 pixels, Tiny-ImageNet is structured into 200 classes, each with 500 training, 50 validation, and 50 test images. This dataset serves as a compact version of ImageNet, catering to visual recognition challenges.
- **ImageNette [9]**: This dataset is a subset of ImageNet, widely used in the research community [29, 31]. It consists of 9,469 training images and 3,925 test images. Each image is with a high resolution of 224×224.

These datasets are widely used to evaluate backdoor attacks' performances on DNNs for computer vision [3, 6, 21] and are also the benchmarking datasets for SOTA backdoor benchmarks and toolboxes [15, 28].

More Details on Backdoor Attacks. In this work, we evaluate 3 existing quantization-conditioned backdoor attacks, *i.e.*, CompArtifact [25], Qu-ANTI-zation [8], and PQBackdoor [18, 19]. Below is the introduction of each attack and their implementation details:

- **CompArtifact [25]**: CompArtifact uses the trigger pattern from BadNets [7], *i.e.*, a 3×3 small white patch on the right lower corner of the image. It is robust to calibration set changes but has low transferability across different bandwidths. Therefore, in our work, we respectively train compromised models for each bandwidth for fair comparison. We use their released official code[1]. Following their original design, we first train a clean model for 400 epochs using standard cross-entropy loss, and then re-train each model (respectively for 8-bit and 4-bit) with the modified objective for 50 epochs, where the poison rate is set to 50% during re-training.
- **Qu-ANTI-zation [8]**: To help the attack transfer, Qu-ANTI-zation considers multiple bit bandwidths in the retraining stage. It showed robustness against several quantization bandwidths as well as robust quantization techniques. It also uses the patch-based trigger, whereas the size is set to 4×4 on CIFAR10 and 8×8 on Tiny-ImageNet. In our evaluation, we use their released official code[2]. Following their original design, we first train a clean model for 200 epochs. Then we re-train the model with the modified objective for 50 epochs, where the poison rate is also set to 50% during re-training.
- **PQBackdoor [18, 19]**: PQBackdoor is the most recent and the SOTA quantization-conditioned backdoor attack. It improves the training pipeline via introducing a two-stage attack strategy: firstly, train a backdoored full-precision model, and secondly, make the backdoor dormant by re-training using the projected gradient descent [20]. This stabilizes the training of the quantization-conditioned backdoor and further improves its robustness. It also uses the patch-based trigger and the size is set to 6×6. PQbackdoor also demonstrated its robustness against blind backdoor defenses such as fine-tuning, and its transferability to commercial quantization frameworks like PyTorch Mobile [22] and TensorFlow Lite [1]. We use the official PyTorch source code from the authors[3] and follow their settings in the paper. For the first stage, the poisoning rate is set to 1%, with the standard training pipeline on poisoning-based backdoor attacks for 100 epochs. After the first stage, the poisoning rate is then set to 50% in the second stage, which takes another 50 epochs. Unfortunately, even if we tried several times (>5), we failed to obtain a full-precision model with CDA reported in their paper. On CIFAR10, we can only have 86.43% with ResNet-18 during our reproduction, much lower than 93.44% reported in their original paper. On Tiny-ImageNet, the CDA is even worse (35.5%), which is much lower than the clean models (usually around 58%). This makes the attacked model less likely to be used by the victim. A possible reason is the network does not fully converge during the first stage (only 100 epochs). To verify this, we train another model for 400 epochs during the first stage and find we can indeed obtain a model with higher precision (93.03% on CIFAR10 and 58.5% on Tiny-ImageNet). To best align with the paper setting and consider the real-world scenarios, in our main paper, we report the results of PQBackdoor with these lower CDA models on CIFAR10 but higher CDA models on Tiny-

[1] https://github.com/yulongt23/Stealthy-Backdoors-as-Compression-Artifacts

[2] https://github.com/Secure-AI-Systems-Group/Qu-ANTI-zation

[3] https://github.com/quantization-backdoor

Table 1. **Defense Results on PQBackdoor (Higher CDA Model) on CIFAR10 (%).** Standard means standard quantization.

Bandwidth	Setting	CDA / ASR
32-bit	Full-precision	93.02 / 9.20
8-bit	Standard	92.49 / 97.43
	EFRAP	90.49 / 7.28
4-bit	Standard	89.79 / 96.49
	EFRAP	89.80 / 0.64

Table 2. **Results on Full-precision Models (%).** For CompArtifact, we train 2 models, respectively for 8-bit attack (8-bit) and 4-bit attack (4-bit). For PQBackdoor (higher CDA), the backdoor model is trained with 400 epochs during the first stage.

Dataset	Attack	CDA / ASR
CIFAR10	Clean Model	93.44% / 0.44%
	CompArtifact (8-bit)	91.46% / 1.26%
	CompArtifact (4-bit)	93.68% / 1.33%
	Qu-ANTI-zation	93.17% / 2.18%
	PQBackdoor	86.43% / 2.67%
	PQBackdoor (higher CDA)	93.02% / 3.20%
Tiny-ImageNet	Clean Model	57.77% / 0.21%
	CompArtifact (8-bit)	57.09% / 0.78%
	CompArtifact (4-bit)	56.89% / 1.43%
	Qu-ANTI-zation	55.82% / 2.16%
	PQBackdoor (higher CDA)	58.50% / 0.88%

ImageNet. We place the defense results for the higher CDA model in Table 1.

More Details on Backdoor Defenses. In this paper, we considered 8 possible defenses against quantization-conditioned backdoor attacks, which are broadly classified into backdoor defense and robust quantization. For backdoor defenses, we consider 5 SOTA backdoor defenses, including FT [24], FP [16], MCR [32], NAD [14], and I-BAU [30]. For all defenses, we use the open-source code from BackdoorBox[4] [15], except for I-BAU, which we use their official implementation[5]. Here are their brief introduction and implementation details:

- **FT [24]:** Fine-tuning (FT) is the most frequently considered baseline for backdoor defenses. It directly fine-tunes the model using a small set of clean data. Though sounds simple, it can effectively remove backdoor effects for many SOTA backdoor attacks [28]. In our work, we fine-tune all layers of the compromised full-precision model using 5% clean data for 50 epochs.
- **FP [16]:** Fine-pruning (FP) is a defense combining fine-tuning and pruning. It first feeds a small set of clean data to the network and measures the activation, then prunes the neurons less frequently activated (which are considered backdoor neurons). To maintain clean accuracy, fine-tuning is involved after pruning. In our work, we measure the activation of the last residual block and the pruning rate is set to 0.4. We then fine-tune the model with 5% clean data for 50 epochs.
- **MCR [32]:** Mode connectivity repair (MCR) is a defense that visits DNN life-cycle security from the loss landscape's perspective. It first fine-tunes a backdoored model, then employs mode connectivity in loss landscapes between the original backdoored model and the fine-tuned model, and finally measures and removes backdoor functions through mode connectivity repair. In our work, we first fine-tune the backdoored model for 50 epochs, then run 100 epochs of curvenet training, and finally 100 epochs of model updating. The hyperparameter t is respectively set to 0.1 and 0.9 and we report the results with higher DTM.
- **NAD [14]:** Neural attention distillation (NAD) is a defense using knowledge distillation with attention guidance. It observes that the attention of backdoored and clean models are different, so it first fine-tunes a backdoored model, which is referred to as a less poisonous model, and then uses this less poisonous model as the teacher model, the original backdoored model as the student model, and conduct knowledge distillation with attention alignment guidance. We run 50 epochs of fine-tuning to obtain the teacher model and 50 epochs to purify the student model.
- **I-BAU [30]:** Implicit backdoor adversarial unlearning (I-BAU) views the task of backdoor removal as a minimax formulation. It then utilizes the implicit hypergradient to account for the interdependence between inner and outer optimization. It is shown faster, more computationally efficient, and more effective than previous backdoor defenses, achieving SOTA defense results on many benchmarks [28]. We run 3 rounds of I-BAU for each attack.

All the aforementioned backdoor defenses have shown effectiveness against SOTA backdoor attacks [15, 28], not to mention the rudimentary backdoor of BadNets [7] used by many quantization-conditioned backdoors. As reported in [28], many evaluated defenses in this paper can reduce the ASR of BadNets to nearly 0% while maintaining high clean accuracy. However, as we show in the main paper, their performances are largely weakened or even ineffective. The main possible reason is that these conditioned backdoors stay dormant on the full-precision models, making the assumption of many backdoor defenses (assuming the existence of explicit backdoors) invalid.

For robust quantization, following Hong et al. [8], OMSE [5], OCS [33], and ACIQ [2] are considered. Here are their brief introduction and implementation details:

- **OMSE [5]:** Optimal MSE (OMSE) is a widely used technique for robust post-training quantization. It formalizes the linear quantization task as a Minimum Mean Squared problem for both weights and activations and solves it via layer-wise optimization. It can largely avoid the severe

[4] https://github.com/THUYimingLi/BackdoorBox

[5] https://github.com/YiZeng623/I-BAU

expected behavioral change of vanilla quantization.

- **OCS [33]:** Outlier channel splitting (OCS) improves quantization via duplicating channels containing outliers and halving the channel values, thus largely avoiding outliers in the distribution. It is shown superior than SOTA clipping techniques with only minor overhead.
- **ACIQ [2]:** Analytical clipping for integer quantization (ACIQ) analytically computes the clipping range as well as the per-channel bit allocation for DNNs, thus enhancing the robustness of model quantization.

Implementation Details. For all experiments, we use Python 3.8.18 and PyTorch 1.10.0+cu113 framework, with torchvision 0.11.1. All experiments are implemented in Python and run on a 14-core Intel(R) Xeon(R) Gold 5117 CPU @2.00GHz with a single NVIDIA GeForce RTX 3090 GPU machine running Linux version 5.4.0-144-generic (buildd@lcy02-amd64-089) (Ubuntu 9.4.0-1 ubuntu1$\tilde{2}$0.04.1). Unless otherwise stated, we use Adam optimizer [10] with default parameters. All other hyperparameters follow the original setting described in the paper. During clean model training and backdoor model training (first stage for PQBackdoor), the learning rate is set to 1e-3, whereas it is set to 1e-4 for all backdoor defenses and the second stage of PQBackdoor. The batch size is set to 64 for CIFAR10 and Tiny-ImageNet, and 16 for ImageNette. Each attack finally results in a full-precision model with a dormant backdoor inserted on each dataset and model architecture. For all experiments, we repeat the experiment at least three times and report the average results in the paper. The standard deviation are small (usually less than 2% for both ASR and CDA). Unless otherwise stated, all activations are also quantized with the same bandwidth of weights. As shown in Table 2, quantization-conditioned backdoors hide well on full-precision models, with a CDA similar to that of a clean model, and an ASR of nearly 0%.

B. More Ablation Studies

B.1. Ablation Study on 8-bit Attacks

Due to the page limit, the ablation study on 8-bit attacks is placed in the Appendix. Except for the evaluated bandwidth, other settings are the same as in the main paper. Here are the ablation results:

Effectiveness of Each Component. As shown in Table 3, on 8-bit attacks, the results keep a similar trend as in 4-bit attacks. Different from 4-bit attacks, the $\mathcal{L}_F$ term alone does not cause severe harm to CDA. This is probably because the 8-bit quantization errors are small and the model learns to be robust to such small flipped rounding errors. However, we can still see that $\mathcal{L}_A$ restores some of the neurons critical for CDA. This further validates the effectiveness of each component proposed in EFRAP.

Effect of Weighting Parameters λ_A and λ_P. As illustrated in Figure 4, on 8-bit attacks, EFRAP is still not sensitive to the choice of weighting parameters on 8-bit settings. This aligns with our conclusion in the main paper.

Table 3. **Ablation Study on Each Component.** $\mathcal{L}'_F$ means $\mathcal{L}_F$ w/o error guidance, *i.e.* do not multiply E when calculating $\mathcal{L}_F$.

Component				8bit Attack
$\mathcal{L}'_F$	$\mathcal{L}_F$	$\mathcal{L}_A$	$\mathcal{L}_P$	CDA ↑ / ASR ↓ / DTM ↑
–	–	–	–	85.16 / 99.11 / 42.58
–	✓	–	–	83.17 / **1.41** / 90.44
–	✓	✓	–	86.15 / 2.03 / 91.62
✓	–	✓	✓	86.06 / 2.99 / 91.09
–	✓	✓	✓	**86.52** / 2.38 / **91.63**

B.2. Ablation Study on Numbers of Iterations

EFRAP optimizes the network layer-by-layer. To better understand the convergence of EFRAP, we examine the influence of the number of iterations by changing the optimization iteration of EFRAP in the layer-wise optimization. The results are shown in Figure 2. We can see that the attack takes effect (ASR<20%) when iteration is above 200, and EFRAP is about to converge with 1000 iterations. To ensure convergence we uniformly take 10000 iterations for our evaluations. This takes about 7 minutes to quantize a ResNet-18 model on Tiny-ImageNet.

C. Resistance to Potential Adaptive Attacks

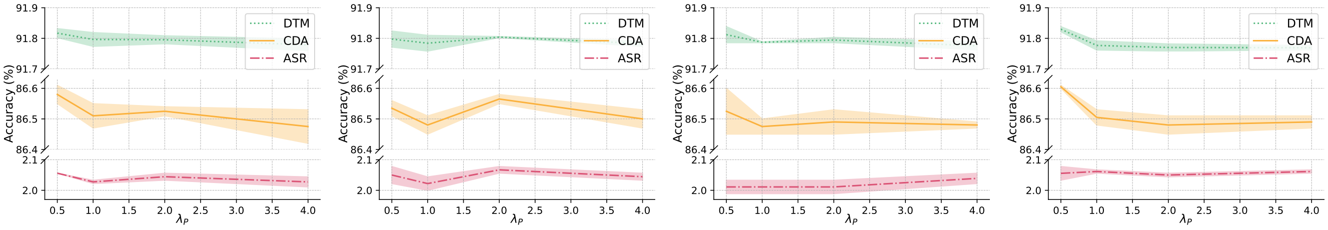
To comprehensively evaluate the robustness of EFRAP, we consider a very smart attacker who is informed of the design of EFRAP and tries to bypass it. According to our threat model, the attacker controls the total training procedure. Thus, he/she can modify the training objective, in order to bypass EFRAP. Specifically, we consider bypassing EFRAP via enforcing the conditioned backdoor to still be activated even if all neurons are flipped rounded. To facilitate a better understanding, in this section, we first present the threat model and the adaptive attack strategy. Then we analyse the effectiveness of the proposed adaptive attacks and give further discussions.

C.1. Threat Model

The threat model for the defender is the same as that of the main paper. As for the attacker, we assume he/she can not only control the whole training dataset as well as the training procedure but is also informed of the design of EFRAP. The attacker aims to bypass the defense via adaptive strategies.

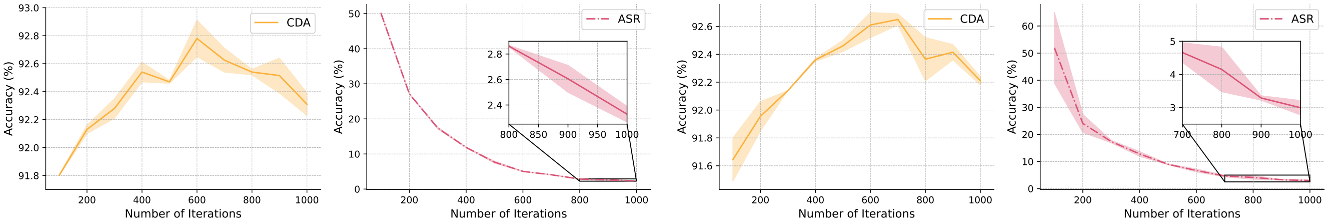
C.2. Attack Methods and Results

Attack Method. EFRAP's effectiveness against backdoors largely relies on the flipped rounding objective, which



(a) 8-bit, $\lambda_A = 0.5$ (b) 8-bit, $\lambda_A = 1$ (c) 8-bit, $\lambda_A = 2$ (d) 8-bit, $\lambda_A = 4$

Figure 1. **Ablation Study on Weighting Parameters.** We repeat each experiment three times.



(a) 8-bit Attack Results (b) 4-bit Attack Results

Figure 2. **Ablation Study on Number of Iterations.** We repeat each experiment three times.

Algorithm 1 Adaptive Attacks (Re-training Stage)

```
Input: A pre-trained clean model $f$ with weights $\boldsymbol{W}$, training set $\mathcal{D}$, quantization scale $s$, learning rate $\tau$.
Output: Backdoored model weights $\boldsymbol{W}$.
while not converged do
    ▷ Record rounding strategy of nearest rounding
    $R(\boldsymbol{W}) \leftarrow \mathbb{1}\{s \cdot \lfloor \frac{\boldsymbol{W}}{s} \rceil - \boldsymbol{W} \succ 0\}$    ▷ Rounding strategy
    $\overline{R}(\boldsymbol{W}) \leftarrow 1 - R(\boldsymbol{W})$    ▷ Flipped rounding strategy
    ▷ Quantization with original and flipped strategy.
    $Q(\boldsymbol{W}) \leftarrow s \cdot clip\left(\left\lfloor \frac{\boldsymbol{W}}{s} \right\rfloor + R(\boldsymbol{W}), n, p\right)$
    $\overline{Q}(\boldsymbol{W}) \leftarrow s \cdot clip\left(\left\lfloor \frac{\boldsymbol{W}}{s} \right\rfloor + \overline{R}(\boldsymbol{W}), n, p\right)$
    ▷ Behave normally on full-precision model.
    $\mathcal{L}_C \leftarrow \mathcal{L}_{ce}(f(\boldsymbol{x}), y) + \alpha \cdot \mathcal{L}_{ce}(f(\boldsymbol{x_t}), y)$
    ▷ Backdoor objectives on nearest rounding.
    $\mathcal{L}_Q \leftarrow \beta \cdot \mathcal{L}_{ce}(f_Q(\boldsymbol{x}), y) + \gamma \cdot \mathcal{L}_{ce}(f_Q(\boldsymbol{x_t}), y_t)$
    ▷ Backdoor objectives on flipped rounding.
    $\mathcal{L}_F \leftarrow \zeta \cdot \mathcal{L}_{ce}(f_{\overline{Q}}(\boldsymbol{x}), y) + \eta \cdot \mathcal{L}_{ce}(f_{\overline{Q}}(\boldsymbol{x_t}), y_t)$
    Get a batch of $\boldsymbol{x}$ from $\mathcal{D}$
    $\mathcal{L} \leftarrow \mathcal{L}_C + \lambda_Q \mathcal{L}_Q + \lambda_F \mathcal{L}_F$
    Update $\boldsymbol{W} \leftarrow \boldsymbol{W} - \tau \cdot \nabla_f \mathcal{L}$
return $\boldsymbol{W}$
```

breaks the connection between rounding errors and backdoor activation. Therefore, an adaptive strategy is to maintain such connection when neurons are flipped, *i.e.*, ensure the backdoor will still be activated even if all neurons are flipped. Specifically, the attacker may implement the adaptive attack by involving a new objective, *i.e.*, maintain backdoor effects on the flipped rounded quantized model.

Experimental Settings. We adopt the training procedure in [8] to implement the adaptive attacks. We first train a clean full-precision model for 400 epochs, then we use the modified training pipeline in Algorithm 1 to re-train the clean model for 50 epochs to insert the quantization-conditioned backdoor. We also tried the training procedure of PQBackdoor [18, 19] (first train a backdoored full-precision model then hide the backdoor using the modified objective with PGD) but the results are similar. All experiments are conducted on CIFAR10 and ResNet-18. The other hyper-parameters and implementation details follow the settings described in Section A.

Results & Analysis. The results of adaptive attacks are in Table 4. On 8-bit attacks, we can see that the attack indeed works well on full-precision models and quantized models. The backdoor hides well in full-precision mode, with a high CDA and low ASR. The adaptive strategy also works well on both standard and flipped rounding strategies, with both high CDA and ASR. However, it fails to defeat EFRAP, where the defended model expresses high CDA and very low ASR. The reason is that EFRAP selectively flips the neurons based on the two objectives, rather than flipping all the neurons. We calculated the ratio of neurons flipped by EFRAP in each layer of a given model and found that the flip rates are varying from model to model and layer to layer, usually between $1\% \sim 40\%$. Therefore, the final rounding strategy of EFRAP is neither nearest rounding nor flipped rounding, making it still effective in breaking the connections between rounding errors and backdoor activations. We also observe the attack results are less satisfactory (CDA=41.36% on Flipped) on the 4-bit setting. This is because the flipped rounding causes larger errors than near-

Table 4. **Results on Adaptive Attacks (%).** Standard means standard quantization and Flipped means quantization with flipped rounding strategy.

Bandwidth	Setting	CDA / ASR
8-bit	Full-precision	93.29 / 0.84
	Standard	93.36 / 100.0
	Flipped	92.12 / 98.57
	EFRAP	92.16 / 1.74
4-bit	Full-precision	93.29 / 0.84
	Standard	88.42 / 100.0
	Flipped	41.36 / 99.88
	EFRAP	92.35 / 1.12

est rounding, especially in the 4-bit setting, which makes it harder to maintain a high CDA.

More Advanced Attacks. Considering the failure of directly implanting backdoors into the flipped rounding strategy, we consider two more advanced adaptive attacks: random flipping and adversarial training with EFRAP. Random flipping refers to randomly flipping some neurons' rounding strategy at each iteration, while the adversarial training with EFRAP refers to conducting EFRAP every single iteration and implanting backdoors into the rounding strategy of EFRAP. These two strategies simulate the possible effect of EFRAP and expect to learn a robust backdoor against it. Note that adversarial training with EFRAP is very time-consuming as conducting EFRAP each time requires about 7 minutes. Therefore, it takes about $50 \times 781 \times 7/60 = 4555$ GPU hours or nearly 190 GPU days to re-train a single ResNet-18 model on CIFAR10 for a re-training stage of 50 epochs, in stark contrast to the original re-training, which takes only 1.5 hours. Therefore, we conduct EFRAP every 50 steps, and the iteration of EFRAP is set to 1000 as a computationally feasible proxy. However, both these strategies failed to bypass EFRAP, even though we tried different flip rates (from 1% to 40%), learning rates, batch size, *etc*., for several times. These attacks all either fail to defeat EFRAP (with a high CDA and very low ASR), or the network can only get bad performances on CDA (usually $< 20\%$). One possible explanation is that such a simulation approach generates unstable rounding strategies and corresponding quantized networks at every step, making it much more challenging to identify a clear and convergent optimization direction than straightforward quantization-conditioned backdoors. Besides, the simulated rounding strategies are still different from EFRAP's final strategy, making the adaptive attack less robust against EFRAP. As the security research on backdoor vulnerabilities is an evolving game between attacks and defenses, we leave the study on more effective attacks to future work.

D. More Visualization Results

In this section, we provide more visualization results, including GradCAM [23] and t-SNE [26].

More GradCAM [23] Results. We provide more GradCAM results for each attack, including CompArtifact, Qu-ANTI-zation, and PQBackdoor, on CIFAR10 and Tiny-ImageNet, and PQBackdoor with advanced trigger (input-aware dynamic and warping-based) on CIFAR10, with 8-bit and 4-bit bandwidth, before and after defense. The results are shown in Figure 4a to 4e. The GradCAM results also demonstrate the effectiveness of EFRAP. After defense, the networks' activation focuses on the main object of the image, rather than the trigger area on the input.

More t-SNE [26] Results. We provide more t-SNE results for each attack, including CompArtifact, Qu-ANTI-zation, and PQBackdoor, on CIFAR10, with 8-bit and 4-bit bandwidth, before and after defense. The results are shown in Figure 5a to 5c. After EFRAP, the poisoned samples effectively disperse to their original category. This shows that EFRAP has successfully removed the backdoor effects in the model.

E. Discussions

Ethical Statements. The study of the security vulnerabilities of deep learning models has the potential to give rise to ethical concerns [4, 17, 27]. In this paper, for the first time, we propose a novel defense against the recently proposed quantization-conditioned backdoor attacks. We are confident that our method will strengthen the security of model quantization process, and safeguard the responsible deployment of deep learning models. We have carefully checked the CVPR 2024 Ethics Guidelines for Authors[6] and we are confident our research adheres to all mentioned ethical standards. We ensure that our methodologies and experiments do not harm individuals or organizations and comply with all relevant ethical guidelines and regulatory standards. Our defense mechanism, EFRAP, is intended solely for protecting DNNs against malicious tampering and is not designed for any unethical or harmful applications.

Statistical analysis to prove neuron's 'dual encoding'. We test the reduction of CDA/ASR after pruning each neuron with top 10% error. As in Fig. 3, they mostly have high relations with ASR, while some of them are also key for CDA, *i.e.*, **some neurons encode both backdoor and clean functions**. This result aligns with Fine-Pruning [16].

More implementation of activation preservation. As suggested by [13], changing layer-wise activation preservation to block-wise can allow a more flexible optimization. We study a case on ResNet-18, PQBackdoor, and indeed a slight improvement (around 0.3% on CIFAR10) is observed. We leave detailed investigations to future work.

[6] https://cvpr.thecvf.com/Conferences/2024/EthicsGuidelines

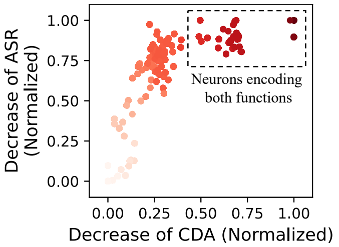


Figure 3. **Neuron Function Study.**

Defenses results using Straight-Through Estimator (STE) on quantized models. In our early trials, we have also considered applying existing defenses on quantized models via STE. However, as in Tab. 5 and 6, the results are still discouraging. The most possible reasons are: (1) STE returns only coarse gradients, not perfectly accurate ones; (2) as the model is already trained with QAT (which already involved STE) by the attacker, the gradients of quantized and full-precision models are similar overall. Therefore the optimization directions are also similar in general, making a significant improvement less likely.

Table 5. **Results w/ STE on 8-bit PQ Backdoor.**

Defense	CDA / ASR
FT + STE	84.38 / 95.41
MCR + STE	84.39 / 65.09
NAD + STE	38.00 / 5.86
I-BAU + STE	82.05 / 19.58

Table 6. **Results w/ STE on 4-bit PQ Backdoor.**

Defense	CDA / ASR
FT + STE	82.95 / 93.12
MCR + STE	82.16 / 40.29
NAD + STE	39.67 / 11.17
I-BAU + STE	76.15 / 20.80

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016. 1

[2] Ron Banner, Yury Nahshan, and Daniel Soudry. Post training 4-bit quantization of convolutional networks for rapid-deployment. In *NeurIPS*, 2019. 2, 3

[3] Ruisi Cai, Zhenyu Zhang, Tianlong Chen, Xiaohan Chen, and Zhangyang Wang. Randomized channel shuffling: Minimal-overhead backdoor attack detection without clean datasets. In *NeurIPS*, 2022. 1

[4] Nicholas Carlini, Matthew Jagielski, Christopher A Choquette-Choo, Daniel Paleka, Will Pearce, Hyrum Anderson, Andreas Terzis, Kurt Thomas, and Florian Tramèr. Poisoning web-scale training datasets is practical. *arXiv preprint arXiv:2302.10149*, 2023. 5

[5] Yoni Choukroun, Eli Kravchik, Fan Yang, and Pavel Kisilev. Low-bit quantization of neural networks for efficient inference. In *ICCVW*, 2019. 2

[6] Tian Dong, Ziyuan Zhang, Han Qiu, Tianwei Zhang, Hewu Li, and Terry Wang. Mind your heart: Stealthy backdoor attack on dynamic deep neural network in edge computing. In *IEEE INFOCOM*, 2023. 1

[7] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *IEEE Access*, 2017. 1, 2

[8] Sanghyun Hong, Michael-Andrei Panaitescu-Liess, Yigitcan Kaya, and Tudor Dumitras. Qu-anti-zation: Exploiting quantization artifacts for achieving adversarial outcomes. In *NeurIPS*, 2021. 1, 2, 4, 7, 8

[9] Jeremy Howard and fastai community. Imagenette. `https://github.com/fastai/imagenette`, 2023. 1

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3

[11] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 1

[12] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 2015. 1

[13] Yuhang Li, Ruihao Gong, Xu Tan, Yang Yang, Peng Hu, Qi Zhang, Fengwei Yu, Wei Wang, and Shi Gu. Brecq: Pushing the limit of post-training quantization by block reconstruction. *arXiv preprint arXiv:2102.05426*, 2021. 5

[14] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Neural attention distillation: Erasing backdoor triggers from deep neural networks. In *ICLR*, 2021. 2

[15] Yiming Li, Mengxi Ya, Yang Bai, Yong Jiang, and Shu-Tao Xia. Backdoorbox: A python toolbox for backdoor learning. *arXiv preprint arXiv:2302.01762*, 2023. 1, 2

[16] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*, 2018. 2, 5

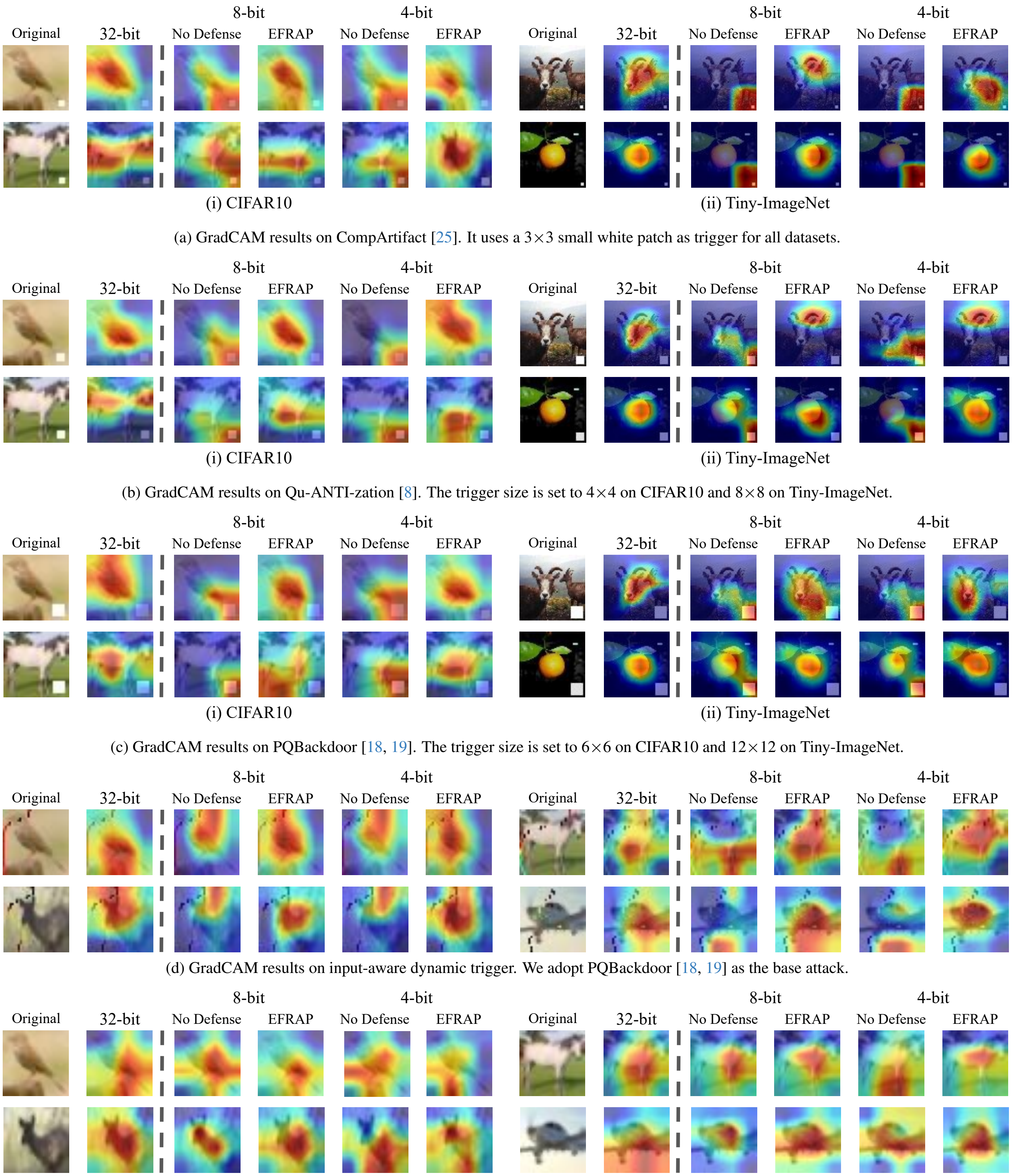
[17] Yingqi Liu, Guangyu Shen, Guanhong Tao, Zhenting Wang, Shiqing Ma, and Xiangyu Zhang. Complex backdoor detection by symmetric feature differencing. In *CVPR*, 2022. 5

[18] Hua Ma, Huming Qiu, Yansong Gao, Zhi Zhang, Alsharif Abuadbba, Anmin Fu, Said Al-Sarawi, and Derek Abbott. Quantization backdoors to deep learning models. *arXiv preprint arXiv:2108.09187*, 2021. 1, 4, 7, 8

[19] Hua Ma, Huming Qiu, Yansong Gao, Zhi Zhang, Alsharif Abuadbba, Minhui Xue, Anmin Fu, Jiliang Zhang, Said F Al-Sarawi, and Derek Abbott. Quantization backdoors to deep learning commercial frameworks. *IEEE TDSC*, 2023. 1, 4, 7, 8

[20] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 1

[21] Dung Thuy Nguyen, Tuan Minh Nguyen, Anh Tuan Tran, Khoa D Doan, and KOK SENG WONG. Iba: Towards irreversible backdoor attacks in federated learning. In *NeurIPS*, 2023. 1



(a) GradCAM results on CompArtifact [25]. It uses a 3×3 small white patch as trigger for all datasets.

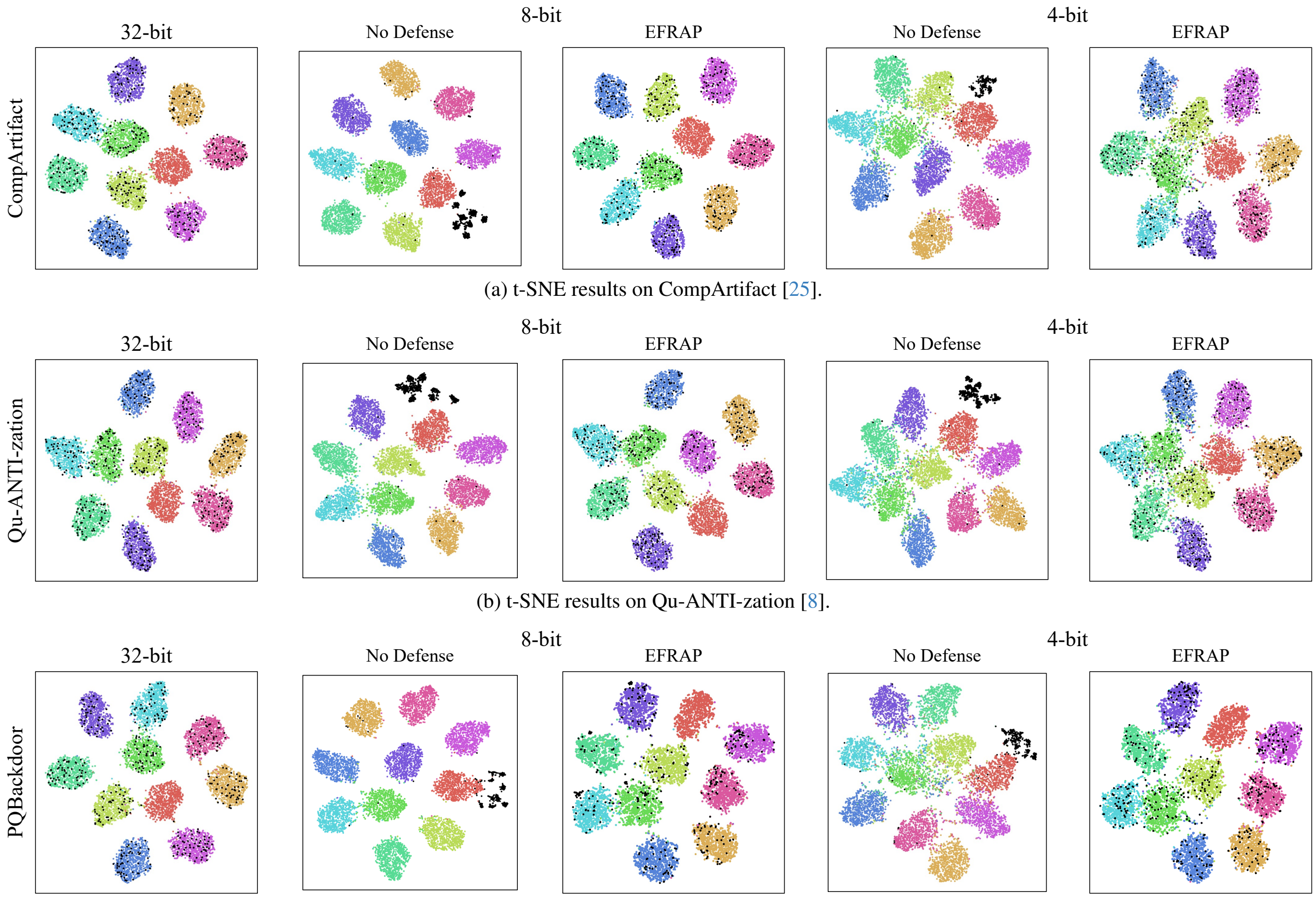
(b) GradCAM results on Qu-ANTI-zation [8]. The trigger size is set to 4×4 on CIFAR10 and 8×8 on Tiny-ImageNet.

(c) GradCAM results on PQBackdoor [18, 19]. The trigger size is set to 6×6 on CIFAR10 and 12×12 on Tiny-ImageNet.

(d) GradCAM results on input-aware dynamic trigger. We adopt PQBackdoor [18, 19] as the base attack.

(e) GradCAM results on warping-based trigger. We adopt PQBackdoor [18, 19] as the base attack.

Figure 4. **More GradCAM Results.**



(a) t-SNE results on CompArtifact [25].

(b) t-SNE results on Qu-ANTI-zation [8].

(c) t-SNE results on PQBackdoor [18, 19].

Figure 5. **More t-SNE Results.**

[22] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 1

[23] 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 *ICCV*, 2017. 5

[24] Zeyang Sha, Xinlei He, Pascal Berrang, Mathias Humbert, and Yang Zhang. Fine-tuning is all you need to mitigate backdoor attacks. *arXiv preprint arXiv:2212.09067*, 2022. 2

[25] Yulong Tian, Fnu Suya, Fengyuan Xu, and David Evans. Stealthy backdoors as compression artifacts. *IEEE TDSC*, 2022. 1, 7, 8

[26] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008. 5

[27] Matthew Walmer, Karan Sikka, Indranil Sur, Abhinav Shrivastava, and Susmit Jha. Dual-key multimodal backdoors for visual question answering. In *CVPR*, 2022. 5

[28] Baoyuan Wu, Hongrui Chen, Mingda Zhang, Zihao Zhu, Shaokui Wei, Danni Yuan, and Chao Shen. Backdoorbench: A comprehensive benchmark of backdoor learning. In *NeurIPS*, 2022. 1, 2

[29] Chong Xiang, Arjun Nitin Bhagoji, Vikash Sehwag, and Prateek Mittal. {PatchGuard}: A provably robust defense against adversarial patches via small receptive fields and masking. In *USENIX Security*, 2021. 1

[30] Yi Zeng, Si Chen, Won Park, Zhuoqing Mao, Ming Jin, and Ruoxi Jia. Adversarial unlearning of backdoors via implicit hypergradient. In *ICLR*, 2022. 2

[31] Yechao Zhang, Shengshan Hu, Leo Yu Zhang, Junyu Shi, Minghui Li, Xiaogeng Liu, Wei Wan, and Hai Jin. Why does little robustness help? a further step towards understanding adversarial transferability. In *IEEE S&P*, 2024. 1

[32] Pu Zhao, Pin-Yu Chen, Payel Das, Karthikeyan Natesan Ramamurthy, and Xue Lin. Bridging mode connectivity in loss landscapes and adversarial robustness. *arXiv preprint arXiv:2005.00060*, 2020. 2

[33] Ritchie Zhao, Yuwei Hu, Jordan Dotzel, Chris De Sa, and Zhiru Zhang. Improving neural network quantization without retraining using outlier channel splitting. In *ICML*, 2019. 2, 3