Error-Silenced Quantization: Bridging Robustness and Compactness[*]

Zhicong Tang , **Yinpeng Dong** and **Hang Su**
Tsinghua University
{tzc17, dyp17}@mails.tsinghua.edu.cn, suhangss@mail.tsinghua.edu.cn

Abstract

As deep neural networks (DNNs) advance rapidly, quantization has become a widely used standard for deployments on resource-limited hardware. However, DNNs are well accepted vulnerable to adversarial attacks, and quantization is found to further weaken the robustness. Adversarial training is proved a feasible defense but depends on a larger network capacity, which contradicts with quantization. Thus in this work, we propose a novel method of Error-silenced Quantization that relaxes the requirement and achieves both robustness and compactness. We first observe the Error Amplification Effect, i.e., small perturbations on adversarial samples being amplified through layers, then a pairing is designed to directly silence the error. Comprehensive experimental results on CIFAR-10 and CIFAR-100 prove that our method fixes the robustness drop against alternative threat models and even outperforms full-precision models. Finally, we study different pairing schemes and secure our method from the obfuscated gradient problem that undermines many previous defenses.

1 Introduction

Deep neural networks (DNNs) have demonstrated extraordinary performances in a wide range of applications, including visual understanding [Krizhevsky *et al.*, 2012; He *et al.*, 2016], speech recognition [Graves *et al.*, 2013], and natural language processing [Devlin *et al.*, 2019]. As its application develops, the deployment of DNNs is becoming omnipresent in embedded and edge devices, such as mobile phones, IoT devices, autonomous driving systems, etc. To facilitate such deployment, quantization [Wu *et al.*, 2016; Jacob *et al.*, 2018] is proposed, which has become an industry standard for deep learning hardware and an accelerator for inference in real-time applications [Rastegari *et al.*, 2016].

However, it is accepted that DNNs are vulnerable to adversarial attacks [Szegedy *et al.*, 2014; Goodfellow *et al.*, 2015], that is, maliciously generated noise hardly noticeable can easily deceive a model to give erroneous predictions. This may lead to disastrous consequences and raises concerns about applications in security-critical domains. For example, in autonomous driving, a stop signal of traffic indicators can be mistakenly detected by a model as a permission signal [Eykholt *et al.*, 2018]; or in face recognition, an adversary can fool the model, bypass the authentication and reach full access to the system [Sharif *et al.*, 2016]. The potential risks are one of the key hindrances to deploy DNNs in safety-critical scenarios.

Furthermore, the commonly used vanilla quantization approaches concentrate on the classification accuracy on clean inputs and may be more severely threatened by adversarial attacks (Table 1). Therefore, it is imperative to develop a quantization algorithm that can jointly optimize robustness and compactness. Adversarial training [Goodfellow *et al.*, 2015; Kurakin *et al.*, 2017; Madry *et al.*, 2018], i.e., augmenting the training set with adversarial samples, is recognized to be one of the best defenses. Nevertheless, it generally requires a significantly larger network capacity than predicting only clean inputs, which is in contradiction to quantization.

To address this issue, we equip quantization with adversarial training and relax the requirement by extracting a pairing object. The pairing of clean and perturbed activation diminishes the error within and is added to the training loss. Then the model concurrently trained and quantized with the loss learns close inference on clean and adversarial inputs and thus achieve both strong robustness and high compactness. Though previous works [Galloway *et al.*, 2018; Gui *et al.*, 2019] are aware of the robustness drop and attempt to fix it, their settings are limited. We thoroughly prove the robustness of our method against four threat models: *white-box attack*, in which attackers have full access to target models; *score and decision based black-box attack*, in which attackers have access to detailed or final predictions; and *transfer attack*, in which attackers know only data distributions.

Experiments demonstrated our contributions: (i) We firstly plotted the precise error in activation of attacked models. (ii) We proposed a novel quantization that directly regulates the perturbed activation. (iii) With the method we silence the error and bridge robustness with model compactness. (iv) We further confirmed the superiority and security of our method. The method is called **Error-silenced Quantization (EQ)** since it is inspired by the Error Amplification Effect and aims at silencing the error in both activation and predictions.

2 Background

2.1 Compress with quantization

In this section, we briefly introduce two typical quantized networks, including Binary Weight Network (BWN) [Rastegari *et al.*, 2016] and Ternary Weight Network (TWN) [Li and Liu, 2016].

Firstly, the weight $\mathcal{W}$ of a DNN can be denoted by $\mathcal{W}_l = \{\mathbf{W}_1, \cdots, \mathbf{W}_i, \cdots, \mathbf{W}_m\}$, where the l-th layer has m output channels and $\mathbf{W}_i \in \mathbb{R}^d$ is the weight of the i-th filter. Quantization converts each weight matrix $\mathbf{W}_i$ into $\mathbf{Q}_i \in \mathrm{S}^d$, where S consists of at most 2^n sparse values in a n-bit quantization.

BWN takes a scaling factor $\alpha \in \mathbb{R}^+$ and $\mathrm{S} = \{-\alpha, +\alpha\}$. By solving the optimization $\mathcal{J} = \min \|\mathbf{W}_i - \alpha \mathbf{B}_i\|$ it yields

$$\mathbf{B}_i^j = \alpha \times \mathrm{sign}(\mathbf{W}_i^j) \quad \text{and} \quad \alpha = \frac{1}{d}\sum_{j=1}^{d}\left|\mathbf{W}_i^j\right|. \tag{1}$$

TWN introduces a 0 state over BWN in $\mathrm{S} = \{-\alpha, 0, +\alpha\}$ to approximate the real-valued weight $\mathbf{W}_i$ more precisely. It solves the optimization $\mathcal{J} = \min \|\mathbf{W}_i - \alpha \mathbf{T}_i\|$ as

$$\mathbf{T}_i^j = \begin{cases} -\alpha, & \mathbf{W}_i^j < -\Delta \\ 0, & \left|\mathbf{W}_i^j\right| \leq \Delta \\ +\alpha, & \mathbf{W}_i^j > \Delta \end{cases} \quad \text{and} \quad \alpha = \frac{1}{|\mathbf{I}_\Delta|}\sum_{i \in \mathbf{I}_\Delta}\left|\mathbf{W}_i^j\right|, \tag{2}$$

where $\Delta = \frac{0.7}{d}\sum_{j=1}^{d}\left|\mathbf{W}_i^j\right|$ and $\mathbf{I}_\Delta = \{j| \left|\mathbf{W}_i^j\right| > \Delta\}$.

Then $\mathbf{B}_i$ and $\mathbf{T}_i$ are the 1-bit and 2-bit quantized $\mathbf{Q}_i$ that forms the space-efficient weight $\mathcal{Q}$. Since the factor α requires little storage, BWN compresses a full-precision model by $\mathbf{32}\times$ and TWN compresses by $\mathbf{16}\times$.

2.2 Adversarial attacks and defenses

Given an image x, adversarial attacks is to find the noise δ that the classifier's prediction of input $x^{\text{adv}} = x + \delta$ is wrong. And defenses aim to maintain the robustness of the classifier, i.e. the prediction accuracy on input x^{adv}. Here we list some attacks and defenses used in experiments.

2.2.1 Attacks

Fast Gradient Sign Method (FGSM) is a L_∞ bounded one-step attack forwarded by [Goodfellow *et al.*, 2015] that calculates the adversarial samples by following the direction of the gradient of loss function L at step size ϵ.

Projected Gradient Descend (PGD) proposed by [Madry *et al.*, 2018] repeats FGSM and starts with a random step to escape the sharp curvature near the original input, and is thought to be the strongest first-order attack.

C&W Attack [Carlini and Wagner, 2017] chooses $\tanh$ function instead of box-constrained methods and optimizes the difference between logits instead of the logit itself. It is an iterative attack and among the strongest L_2 attacks.

Decoupling Direction and Norm Attack (DDN) [Rony *et al.*, 2019] is a newly proposed L_2 attack that outperforms C&W. It iterates FGSM with the ϵ adjusted in each round, leading to a finer-grained search for adversarial images.

ϵ	1	2	4	8	16
NAT-Full	36.19	27.96	20.76	14.53	7.79
NAT-VQ-BWN	35.38	21.07	11.79	7.59	4.99
ADV-Full	47.22	43.65	36.63	24.60	11.16
ADV-VQ-BWN	40.84	28.34	19.00	12.74	7.74

Table 1: Results on **CIFAR-100** and **ResNet-152** support that quantization undermines robustness and the accuracy (in %) of quantized BWN models drops rapidly as ϵ increases. Abbreviations: **NAT-** for naturally trained, **ADV-** for adversarially training, **-VQ-** for vanilla quantization, **-Full** for full precision, **-BWN** for binary weight.

2.2.2 Defenses

Adversarial training [Goodfellow *et al.*, 2015; Kurakin *et al.*, 2017; Madry *et al.*, 2018] is currently the strongest and most commonly used defense. It augments the training set with adversarial samples by the optimization as

$$\min_\theta \mathbf{E}_{(x,y)\sim D}\left[\max_{\delta \in \Delta} L(\theta, x + \delta, y)\right], \tag{3}$$

where pairs of example $x \in \mathbb{R}^d$ and ground-truth y follow an underlying data distribution D, $\delta \in \Delta$ is the allowed adversarial noise added to image x to deceive the classifier, θ is the model weight to be optimized and L is the loss function.

3 The Error Amplification Effect

The conventionally quantized DNN is counter-intuitively more vulnerable [Lin *et al.*, 2019] under the threat of adversarial attacks. One convincing explanation is *the Error Amplification Effect* discovered by [Liao *et al.*, 2018]. Specifically, tiny perturbations can be amplified when fed through layers, become sizable enough to deceive the network and eventually push the classification result into an incorrect bucket. Moreover, the quantization of a DNN worsens its robustness comparing with the original full-precision one by enlarging the granularity of the weights, making its response more susceptible to the input. As shown in Table 1, quantized models yield constantly inferior robustness under FGSM attacks of varied perturbation strength.

To in detail investigate the effect, we conducted pre-experiments on CIFAR-100 [Krizhevsky and Hinton, 2009] and ResNet-152 [He *et al.*, 2016]. Adversarial samples are generated untargeted by a 10-step PGD attacker with other parameters $\epsilon = 8/255$ and step size $2/255$ corresponding to [Madry *et al.*, 2018]. In Figure 1, we test four settings with the attack, evaluate and plot the distance D_l between the clean and perturbed activation of each layer as

$$D_l(x, x^{\text{adv}}) = \frac{\left\|\mathbf{F}_l(x) - \mathbf{F}_l(x^{\text{adv}})\right\|_2}{\|\mathbf{F}_l(x)\|_2}, \tag{4}$$

where $\mathbf{F}_l$ denotes the activation after the l-th ResNet module. For convenience, we note training scheme with prefix NAT- and ADV-, quantization scheme with infix -VQ- and -EQ-, weight precision with suffix -Full, -BWN, -TWN and use acronyms in all tables.

In the left zone of the illustration 1, the adversarial noise applied to the input image is relatively small compared to the

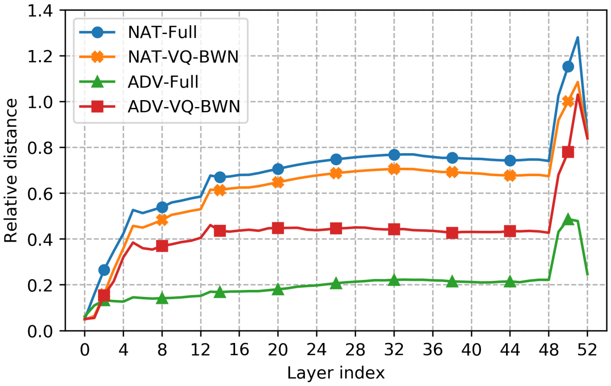


Figure 1: Small perturbations amplified throughout layers and two quantized BWN models predict the same level of error as the undefended naturally trained model. Abbreviations: **NAT-** for naturally trained, **ADV-** for adversarially training, **-VQ-** for vanilla quantization, **-Full** for full precision, **-BWN** for binary weight.

image itself (±8 versus 255 in this setting). However, as the inference carries on the magnitude of initial perturbation is amplified through the latter part of the network. Once the perturbation is amplified large enough, the model is misled to a wrong bucket and the accuracy is witnessed a harsh drop.

With the experiment results above we have the following observations: (i) The error of the activation eventually accumulates large enough to push the prediction to a misleading bucket. (ii) All models suffer from the effect while quantization reduces robustness by a wide margin. (iii) With vanilla quantization methods, the robustness gain of adversarial training is drastically degraded.

Therefore, the currently used vanilla quantizations are showed practically limited and the Error Amplification Effect may be a key to a robustness-aware quantization.

4 Method

Motivated by the Error Amplification Effect above, we introduce a quantization scheme that simultaneously preserves the robustness of the original full-capacity model and the compactness of low bandwidth quantization. The concurrent training and quantizing procedure is described in (1).

We firstly follow the commonly used min-max based robustness optimization and formulate the overall robustness and compactness target as

$$\min_{\theta} \mathbf{E}_{(x,y)\sim D}\left[\max_{\delta\in\Delta} L(\theta, x+\delta, y)\right] \quad s.t. \quad \frac{\text{size}(\theta_{\text{full}})}{\text{size}(\theta)} = c, \tag{5}$$

where θ_{full} is the original full-precision weight ($\mathcal{W}$), θ is the finally quantized weight ($\mathcal{Q}$), size$(\cdot)$ is the memory size to store the weight and c is the target compression rate.

The equation (5) can be divided into two parts: (i) Minimize the loss on adversarially perturbed inputs for robustness. (ii) Compress the model weight to meet the target rate for compactness. In our method, the latter one is handled by a quantization algorithm that allows simultaneous training, and the former one is handled by directly controlling the amplified error, i.e., pairing activation.

Algorithm 1 Error-silenced Quantization

Input: dataset D, full-precision weight θ_{full}, selected layers S and loss function L
Parameter: quantization iteration K, PGD perturbation strength ϵ, PGD iteration T, sensitivity parameters λ_l and distance functions D_l for each layer l
Output: quantized weight θ

```
for $k = 1, 2, \cdots, K$ do
    Sample batch $(x, y)$ from $D$
    Partially quantize $\theta_{\text{full}}$ into $\theta$
    for $t = 1, 2, \cdots, T$ do
        Solve the inner max of Eq (6) to obtain $\delta$
    end for
    $L := L(\theta, x, y)$
    for layer $l$ in $S$ do
        $L = L + \lambda_l D_l(x, x + \delta)$
    end for
    Backward and update $\theta_{\text{full}}$ with loss $L$
end for
return $\theta$
```

4.1 Pairing activation

Since the activation of an adversarial input deviates largely from that of its original image, a natural solution to control the error is training the network to diminish this deviation.

Let $D_l(x, x')$ be a function that calculates the relative distance between the activation of l-th layer when the model is fed with x and x' respectively, which can be normalized L_2 or L_∞. With a set of layers to control S, the robustness regularization that optimizes the former part of (5) is

$$\min_{\theta} \mathbf{E}_{(x,y)\sim D}\left[L(\theta, x, y) + \max_{\delta\in\Delta} P(x, x+\delta)\right]. \tag{6}$$

Here L is the loss function and P is the pairing defined as

$$P(x, x^{\text{adv}}) = \sum_{l\in S} \lambda_l D_l(x, x^{\text{adv}}), \tag{7}$$

where λ_l is a series of sensitivity parameters that determine the threshold of the amplified error between clean and adversarial samples. The model is forced to infer close activation on l-th layer if λ_l is large and is allowed to tolerate sizable differences if λ_l is small.

With the pairing object, we train the model with clean samples and then pair the activation of particular layers, rather than directly training on adversarial samples. The equation (6) can also be divided into two parts that separately tackle the classification accuracy on clean and adversarial images.

The first part is designed to maintain the performance of the model because it is noticed that the development of robustness is often at the cost of prediction accuracy [Su *et al.*, 2018]. With the second part, we train the model to diminish the deviation and infer close activation. A model behaves closely on clean and adversarial inputs is supposed to gain close prediction accuracy on both.

As a special case, pairing is applied only on the final output layer of the network, on which the following experiments focus. Then the pairing can be simplified as the distance between the logits on clean and adversarial samples.

4.2 Solving adversarial perturbations

In the optimization (6), the perturbations δ are generated to maximize the error of selected activation. However, in this work we generate them with untargeted white-box attacks because it is believed the strongest attack and so far no attack studies and magnifies the error.

Previous works [Madry *et al.*, 2018] have shown that PGD performs as the most powerful first-order attack. We follow the conclusion and solve adversarial perturbations δ by PGD attacks with settings consistent with [Madry *et al.*, 2018] and modify iteration number and step size.

4.3 Progressive quantization

Our method upholds and improves the robustness of quantized models by concurrently updating and quantizing its weight. Accordingly, we choose the Stochastic Quantization method introduced in [Dong *et al.*, 2019]. In our method, a model is fed of clean and adversarial inputs with partially quantized weight, and the full-precision weight is updated by the gradients estimated. For comparison, vanilla Stochastic Quantization trains models with clean inputs only.

5 Experiments

In this section, our experiments demonstrate that the proposed method can effectively retain and further improve the robustness when a model is quantized into low-bandwidth. Also, the method diminishes the aforementioned Error Amplification Effect by a large margin compared with both full-precision and vanilla quantized models. Finally, we show that the method provides more convincing performances than two baselines: adversarial training before and while quantization.

5.1 Settings

We apply Wide ResNet 28-10 [Zagoruyko and Komodakis, 2016] on CIFAR-10 [Krizhevsky and Hinton, 2009] and ResNet-152 on CIFAR-100. Six models in each setting are tested with clean input, white-box and transfer attacks.

During training, we augment training set with the PGD attacker same as above and train models with an Adam optimizer [Kingma and Ba, 2015] for 150 epochs. The hyper-parameters are left in default without fine-tuning.

During quantization, we pair the activation after the final layer (logits) by L_2 norm and use a SGD optimizer with learning rate 0.1, momentum 0.9 and weight decay 10^{-4} to train for 120 epochs in consistence with [Dong *et al.*, 2019]. However, the quantization ratio is updated by the uniform scheme, i.e., beginning at 0.2 and updated by 0.2 for every 25 epochs.

5.2 Retaining robustness of quantized models

For white-box attack tests, we use a 20-step PGD attacker with step size 0.1, which is slightly stronger than that used for training. We also analyze the robustness against other adversarial attacks, using $\epsilon = 16/255$ FGSM to study one-step attacks, 100-step $\epsilon = 1$ DDN and 20-step $\epsilon = 1$ C&W to study L_2 bounded attacks.

	NF	NEB	AF	AVB	AEB	AET
Clean	93.33	79.35	80.10	90.84	82.19	81.31
FGSM	7.24	26.47	29.47	22.81	29.49	26.72
PGD	0.00	41.84	47.06	7.08	41.62	41.02
DDN	0.00	29.11	28.18	2.43	28.04	24.81
C&W	0.04	38.58	40.49	8.45	38.24	36.84

(a) **Natural test** and **white-box attack** accuracy (in %). Underline indicates the first and the second of the row.

	NF	NEB	AF	AVB	AEB	AET
NF	0.00	77.68	78.06	77.74	81.11	79.62
NEB	69.10	41.82	60.84	65.58	64.19	64.08
AF	67.44	57.33	47.71	54.49	61.20	60.89
AVB	24.82	73.51	72.75	7.11	76.09	75.31
AEB	75.79	62.74	63.12	64.98	41.36	60.66
AET	77.20	63.31	63.70	67.79	61.11	41.12

(b) **Transfer attack** accuracy (in %). Attacks are generated by row and applied by line, for example, AF model reaches an accuracy of 60.84% on adversarial inputs generated with NEB model.

Table 2: Test results on **CIFAR-10**. Abbreviations: **N-** for naturally training, **A-** for adversarially training, **-V-** for vanilla quantization, **-E-** for Error-silenced Quantization, **-F** for full precision, **-B** for binary weight, and **-T** for ternary weight.

For transfer attack tests, all adversarial samples are generated by the same PGD attacker as white-box stage. We train and quantize alternative models from scratch if the model setting generating attacks and being attacked is the same.

5.2.1 Results

As shown in Table 2a and 3a, the vanilla quantized models are exposed with weak robustness and adversarial training before quantization helps little. With conventional methods, the robustness gained by adversarial training is drastically degraded to nearly none. While with our method, the accuracy consistently floats around or above full-precision models throughout two datasets. Comparing to the gap of vanilla quantization, our proposed method is proved to be feasible in controlling the harsh drop to a reasonably small level and works for both naturally and adversarially trained models.

In the cross transfer attack scenario (Tables 2b and 3b), our robustly quantized models achieve sound results. For adversarial attacks generated from NF models, which is often the situation, the proposed method assists quantized models to steadily beat the AF model. It is also true that our method established solid defenses confronting other attacks, for example, in Table 3b the -EQ- models exceed the AF model under the attacks of other quantized models.

We also notice that the NEB model and the AEB model perform almost the same, which further demonstrates the advantages of our method that adversarial training before quantization is not required. Lastly, the method manages to maintain and even improve accuracy on clean data.

	NF	NEB	AF	AVB	AEB	AET
Clean	73.20	55.54	50.80	65.84	54.09	50.74
FGSM	7.77	12.05	11.15	7.59	13.36	10.78
PGD	0.03	19.17	22.15	0.65	20.49	19.03
DDN	0.01	12.35	17.37	0.24	13.74	12.22
C&W	0.34	18.48	20.62	1.21	19.83	17.02

(a) **Natural test** and **white-box attack** accuracy (in %). Underline indicates the first and the second of the row.

	NF	NEB	AF	AVB	AEB	AET
NF	0.09	52.87	48.63	33.14	52.07	48.57
NEB	49.88	18.88	36.80	41.44	37.69	36.52
AF	44.78	37.27	22.38	33.68	35.75	34.62
AVB	13.77	51.94	46.64	0.57	50.56	47.18
AEB	51.14	37.99	36.31	41.40	20.71	36.45
AET	56.34	40.05	37.50	46.31	39.13	18.67

(b) **Transfer attack** accuracy (in %). Attacks are generated by row and applied by column, for example, AF model reaches an accuracy of 36.80% on adversarial inputs generated with NEB model.

Table 3: Test results on **CIFAR-100**. Abbreviations: **N-** for naturally training, **A-** for adversarially training, **-V-** for vanilla quantization, **-E-** for Error-silenced Quantization, **-F** for full precision, **-B** for binary weight, and **-T** for ternary weight.

5.3 Silencing the Error Amplification Effect

We re-evaluate the error in latent layers to investigate whether the method manages to silence it. The relative distance is defined in (4) and sampled after every ResNet module. The experiment is conducted on ResNet-152 and CIFAR-100.

5.3.1 Results

Though the input is perturbed by the same magnitude, the error is amplified quite differently in Figure 2. With conventional quantization, the error of the ADV-VQ-BWN model increases up to 4 times of the ADV-Full model, which is a possible explanation of the large robustness drop. While with our method, the models managed to lower the error than its full-precision counterpart throughout the inference.

[Xu *et al.*, 2018] conclude that image quantization, i.e., reduction in color bit depth is an effective defense. However, quantization of network weight instead weakens robustness. [Lin *et al.*, 2019] proved that it tends to intensify the Error Amplification Effect when $\epsilon > 3/255$, which even starts from $\epsilon = 1/255$ in our experiments (Table 1). Our method obtains significant results, overcomes the threshold and further pushes it beyond $\epsilon = 8/255$ as in Figure 2.

5.4 Beyond standalone adversarial training

To prove the necessity of pairing, we append experiments of adversarial training in vanilla quantization on ResNet-152 and CIFAR-100.

For adversarial training in vanilla quantization, models are fed with perturbed samples only and updated by the original min-max optimization. All adversarial samples are generated with the same PGD attacker as in the white-box section and all models are quantized for 120 epochs.

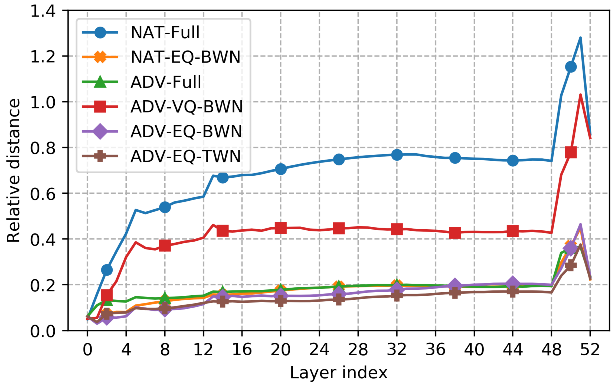


Figure 2: Our quantized models diminish the Error Amplification Effect by a large margin and even outperform full-precision models. Abbreviations: **NAT-** for naturally trained, **ADV-** for adversarially training, **-VQ-** for vanilla quantization, **-EQ-** for Error-silenced Quantization, **-Full** for full precision, **-BWN** for binary weight, and **-TWN** for ternary weight.

Training	Clean	FGSM	PGD	DDN	C&W
Natural	56.39	11.26	19.53	12.55	18.33
Adversarial	49.64	10.07	16.66	10.80	16.10
Natural	54.88	10.16	17.99	10.23	16.30
Adversarial	50.51	9.87	17.63	11.70	16.40

Table 4: Robustness of adversarial training in vanilla quantization. Test accuracy in %. Models are quantized to **1-bit** and **2-bit** in the upper and lower part.

5.4.1 Results

As in the upper part of Table 4, adversarial training in vanilla quantization retains limited robustness and is not comparable to our method. For naturally trained models, adversarial training promotes robustness to 19% against PGD but lags 1% behind our method. For adversarially trained models, adversarial training fails to maintain the robustness and leaves a drop of 5.5%, which is the triple of ours.

We hold that the following hypothesis may lead to the inconsistent performances of adversarial training in the context of ordinary training and quantization: (i) Quantization limits the capacity of the model, while adversarial training requires a significantly large capacity. (ii) With limited capacity, the model faces difficulty in learning and therefore suffers from lower accuracy on both clean and adversarial inputs. In contrast, the model learns to predict only clean inputs and infer close activation on adversarial inputs with our method.

We apply additional experiments on 2-bit quantization to demonstrate the hypothesis above. Though TWN models learn higher accuracy on training set, which confirms our hypothesis that adversarial training is hindered by limited network capacity, they attain the same and even inferior results on test set compared to BWN models. It draws conclusion that while higher bandwidth enables adversarial training, it itself undermines robustness ([Lin *et al.*, 2019]). In contrast, our method better balances the trade-off between adversarial training and low bandwidth weight.

Pairing	Clean	FGSM	PGD	DDN	C&W
Logit	54.09	13.18	20.31	12.20	19.70
Activation	49.65	11.80	18.01	13.10	19.70
Logit	50.74	10.78	19.03	11.20	16.90
Activation	49.54	10.26	18.37	11.04	16.18

Table 5: Robustness of EQ with different pairing target. Test accuracy in %. Models are quantized to **1-bit** and **2-bit** in the upper and lower part.

6 Discussions

In this section, we discuss the equivalence of different pairing scheme and assume pairing logits as a universal pairing. We also discuss the obfuscated gradients problem which undermines many previous defenses and further secure the robustness of our method.

6.1 Equivalence of different pairing

While we offer a general pairing object in (6) and (7) that can be any layers, only the output logits is paired in experiments. Here we reveal that though pairing the activation may produce lower errors, pairing the logits achieves the same accuracy and better balances training costs and performances. We investigate ResNet-152 on CIFAR-100 and pair the activation after the 4th, 12th and 48th ResNet module.

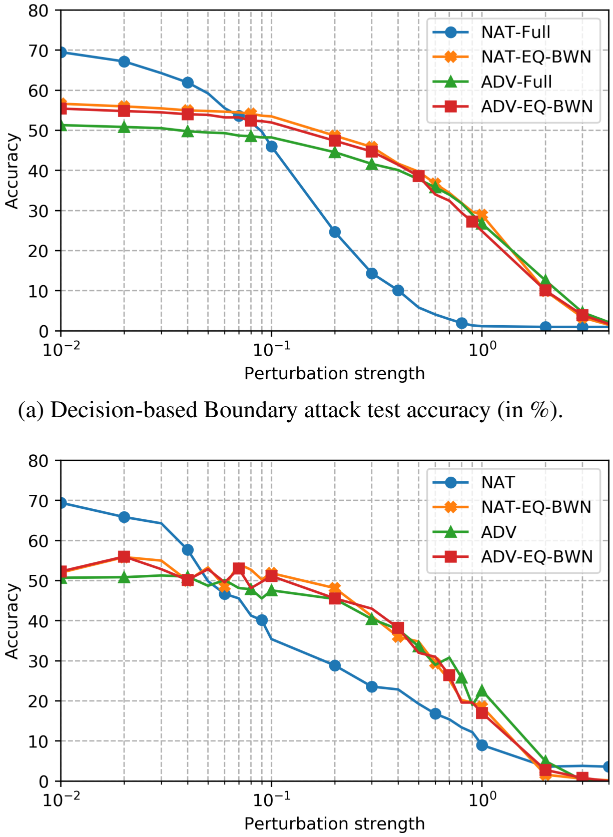
In Table 5, the close accuracy of two pairing schemes shown confirms that pairing more activation provides minor improvements while it requires considerable additional computations and storage of intermediate results. It brings a large cost of memory space, especially when training with GPU. Furthermore, pairing activation may introduce unnecessary requirements on network capacity, as in the case of adversarial training. The smaller gap between two pairing settings on TWN is also an implication of it.

6.2 Secure the sense of robustness

A noticeable coincidence is that our simplified activation pairing scheme, pairing logits, is considerably similar to the *Adversarial Logit Pairing* forwarded in [Kannan *et al.*, 2018]. With the method, the author claims state-of-the-art robustness on ImageNet. However, it is found [Athalye *et al.*, 2018] to suffer severely from obfuscated gradients and provide a false sense of security that can be easily circumvented with non gradient-based attacks.

In [Athalye *et al.*, 2018], it is reported that defenses suffering from obfuscated gradients are vulnerable to black-box attacks that operate by estimating instead of directly solving gradients. To thoroughly examine whether our method is truly secure, we test it with L_2 bounded Boundary attack [Brendel *et al.*, 2018] and $\mathcal{N}$Attack [Li *et al.*, 2019] for decision-based and score-based black-box attacks, respectively. We vary perturbation strength from $\epsilon = 0$ to $\epsilon = 4$ and compare the accuracy of quantized models with full-precision counterparts.

As shown in Figure 3a and 3b, our quantization achieve consistently close or better than the ADV-Full model as the strength varies. All results confirm that our method meets no obfuscated gradient problem and provides a secured sense of robustness. We suppose a possible explanation that we use untargeted attacks for training while [Kannan *et al.*, 2018] use targeted attacks.



(a) Decision-based Boundary attack test accuracy (in %).

(b) Score-based $\mathcal{N}$Attack test accuracy (in %).

Figure 3: Black-box attack test results on **CIFAR-100**. Abbreviations: **NAT-** for naturally trained, **ADV-** for adversarially training, **-EQ-** for Error-silenced Quantization, **-Full** for full precision, **-BWN** for binary weight.

7 Conclusion

This paper aims to tackle the issue of achieving both robustness and compactness in DNNs. Inspired by the Error Amplification Effect, we relax the capacity requirements of adversarial training by pairing, and propose a quantization that optimizes accuracy on benign and adversarial inputs simultaneously. Extensive experiments throughout four threat models, two datasets and two networks endorse the superior robustness of the proposed method over vanilla approaches and even full-precision counterparts, while still reach high compression rates. Appended by a guarded notion of secure from obfuscated gradients, our method managed to bridge robustness and compactness for DNNs and further applications.

References

[Athalye *et al.*, 2018] Anish Athalye, Nicholas Carlini, and David A. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, volume 80, 2018.

[Brendel *et al.*, 2018] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. In *ICLR*. OpenReview.net, 2018.

[Carlini and Wagner, 2017] Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, 2017.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[Dong *et al.*, 2019] Yinpeng Dong, Renkun Ni, Jianguo Li, Yurong Chen, Hang Su, and Jun Zhu. Stochastic quantization for learning accurate low-bit deep neural networks. *International Journal of Computer Vision*, 2019.

[Eykholt *et al.*, 2018] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *CVPR*, 2018.

[Galloway *et al.*, 2018] Angus Galloway, Graham W. Taylor, and Medhat Moussa. Attacking binarized neural networks. In *ICLR*. OpenReview.net, 2018.

[Goodfellow *et al.*, 2015] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

[Graves *et al.*, 2013] Alex Graves, Abdel-rahman Mohamed, and Geoffrey E. Hinton. Speech recognition with deep recurrent neural networks. In *ICASSP*, 2013.

[Gui *et al.*, 2019] Shupeng Gui, Haotao Wang, Haichuan Yang, Chen Yu, Zhangyang Wang, and Ji Liu. Model compression with adversarial robustness: A unified optimization framework. In *NeurIPS*, 2019.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[Jacob *et al.*, 2018] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew G. Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *CVPR*, 2018.

[Kannan *et al.*, 2018] Harini Kannan, Alexey Kurakin, and Ian J. Goodfellow. Adversarial logit pairing. *CoRR*, abs/1803.06373, 2018.

[Kingma and Ba, 2015] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[Krizhevsky and Hinton, 2009] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *University of Toronto, Tech. Rep*, 2009.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[Kurakin *et al.*, 2017] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *ICLR*. OpenReview.net, 2017.

[Li and Liu, 2016] Fengfu Li and Bin Liu. Ternary weight networks. In *NIPS workshop on EMDNN*, 2016.

[Li *et al.*, 2019] Yandong Li, Lijun Li, Liqiang Wang, Tong Zhang, and Boqing Gong. NATTACK: learning the distributions of adversarial examples for an improved black-box attack on deep neural networks. In *ICML*, 2019.

[Liao *et al.*, 2018] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *CVPR*, 2018.

[Lin *et al.*, 2019] Ji Lin, Chuang Gan, and Song Han. Defensive quantization: When efficiency meets robustness. In *ICLR*. OpenReview.net, 2019.

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

[Rastegari *et al.*, 2016] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *ECCV*, 2016.

[Rony *et al.*, 2019] Jérôme Rony, Luiz G. Hafemann, Luiz S. Oliveira, Ismail Ben Ayed, Robert Sabourin, and Eric Granger. Decoupling direction and norm for efficient gradient-based L2 adversarial attacks and defenses. In *CVPR*, 2019.

[Sharif *et al.*, 2016] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K. Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *ACM CCS*, 2016.

[Su *et al.*, 2018] Dong Su, Huan Zhang, Hongge Chen, Jinfeng Yi, Pin-Yu Chen, and Yupeng Gao. Is robustness the cost of accuracy? - A comprehensive study on the robustness of 18 deep image classification models. In *ECCV*, 2018.

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

[Wu *et al.*, 2016] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng. Quantized convolutional neural networks for mobile devices. In *CVPR*, 2016.

[Xu *et al.*, 2018] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. In *NDSS*, 2018.

[Zagoruyko and Komodakis, 2016] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016.