IBD-PSC: Input-level Backdoor Detection via Parameter-oriented Scaling Consistency

Linshan Hou [1] **Ruili Feng** [2 3] **Zhongyun Hua** [1] **Wei Luo** [4] **Leo Yu Zhang** [5] **Yiming Li** [6]

Abstract

Deep neural networks (DNNs) are vulnerable to backdoor attacks, where adversaries can maliciously trigger model misclassifications by implanting a hidden backdoor during model training. This paper proposes a simple yet effective input-level backdoor detection (dubbed IBD-PSC) as a 'firewall' to filter out malicious testing images. Our method is motivated by an intriguing phenomenon, *i.e.*, parameter-oriented scaling consistency (PSC), where the prediction confidences of poisoned samples are significantly more consistent than those of benign ones when amplifying model parameters. In particular, we provide theoretical analysis to safeguard the foundations of the PSC phenomenon. We also design an adaptive method to select BN layers to scale up for effective detection. Extensive experiments are conducted on benchmark datasets, verifying the effectiveness and efficiency of our IBD-PSC method and its resistance to adaptive attacks. Codes are available at BackdoorBox.

1. Introduction

Backdoor attacks are an emerging training-phase threat to deep neural networks (DNNs) (Li et al., 2022b). A backdoored model behaves normally on benign samples while misclassifying malicious samples containing adversary-specified patterns (*i.e.*, triggers). This attack could happen whenever the training stage is not fully controlled. It poses a significant threat to the lifecycle and supply chain of DNNs.

[1]School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China [2]Alibaba Group, China [3]University of Science and Technology of China, China [4]School of Information Technology, Deakin University, Australia [5]School of Information and Communication Technology, Griffith University, Australia [6]Nanyang Technological University, Singapore. Correspondence to: Zhongyun Hua <huazhongyun@hit.edu.cn>, Yiming Li <liyiming.tech@gmail.com>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

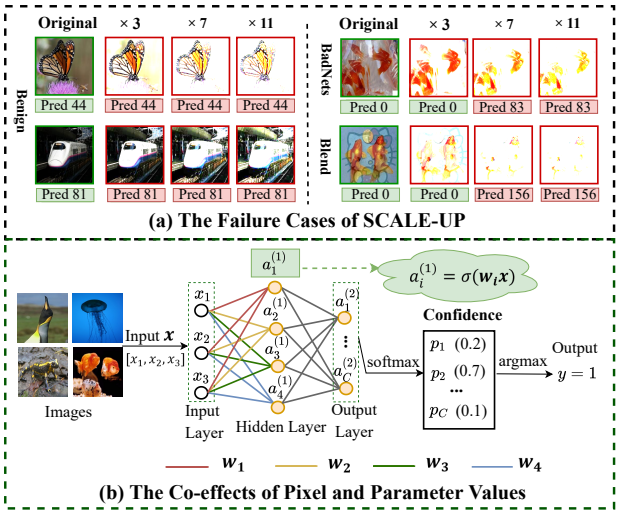


Figure 1. The limitation of SCALE-UP and the co-effects of pixel and parameter values. **(a)** Failures in SCALE-UP due to bounded pixel value (*i.e.*, [0, 255]). Specifically, benign samples with black and white pixels are immune to amplification, preserving scaled prediction stability. Multiplying larger pixel values can easily turn them white, making the trigger disappear and become useless. **(b)** The prediction is the co-effects of the image and model parameters.

Currently, there are five representative defense strategies to alleviate backdoor threats, including **(1)** data purification (Tran et al., 2018; Li et al., 2021b; Jebreel et al., 2023), **(2)** poison suppression (Wang et al., 2022a; Huang et al., 2022; Tang et al., 2023), **(3)** model-level backdoor detection (Wang et al., 2019; Xiang et al., 2023; Wang et al., 2024), **(4)** model-level backdoor mitigation (Liu et al., 2018; Zeng et al., 2022; Guo et al., 2023a), and **(5)** input-level backdoor detection (IBD) (Gao et al., 2021; Liu et al., 2023; Guo et al., 2023b). In general, the first four strategies typically demand substantial computational resources since they usually require model training. However, these resources are unavailable for many researchers and developers, especially those using third-party models. In contrast, the last one is less resource-intensive and is, therefore, our main focus. It aims to detect and prevent malicious inputs and can serve as the firewall of deployed models.

To the best of our knowledge, SCALE-UP (Guo et al., 2023b) currently stands as the most advanced IBD. It ob-

serves that the predictions of poisoned samples (*i.e.*, those containing triggers) exhibit more robustness to pixel-level amplification compared with those of benign samples and provides the theoretical foundations for this phenomenon. Employing this intriguing phenomenon, SCALE-UP directly enlarges all pixel values of the suspicious input sample with varying amplification intensities and assesses its prediction consistency for detection. However, SCALE-UP encounters some intrinsic limitations due to the restriction of pixel values (*i.e.*, bounded in [0, 255]). For example, as shown in Figure 1(a), benign samples containing black and white pixels maintain their initial predictions during the amplification process. This stability is due to their extreme pixel values (0 or 255), which remain unaffected against amplification. Conversely, in poisoned samples, amplification often turns higher pixel values to the maximum (*i.e.*, 255). It leads to large blank areas in the scaled poisoned images, masking the triggers and thus leading to changes in their predictions. Recognizing that prediction results are from the co-effects of pixel and parameter values, as shown in Figure 1(b), while parameter values are not bounded, an intriguing question arises:

Shall the model's parameters expose backdoors with more grace than the humble pixel's tale?

Fortunately, the answer is yes! In this paper, we reveal that the prediction confidences of poisoned samples have *parameter-oriented scaling consistency (PSC)*. Specifically, we scale up the learned parameters of the batch normalization (BN) layers, which are widely exploited in advanced DNN structures. We demonstrate that the prediction confidences of poisoned samples are significantly more consistent than those of benign ones when the number of amplified BN layers increases. In particular, we show that this intriguing phenomenon is not accidental, where we prove that we can always find a scaling factor for BN parameters to expose latent backdoors for all attacked models (under some classical assumptions in learning theory). The scaled model can misclassify benign samples while maintaining the predictions of poisoned samples, leading to the PSC phenomenon.

Motivated by this finding, we propose a simple yet effective IBD method to identify and filter malicious testing samples, dubbed IBD-PSC. Specifically, for each suspicious testing image, our IBD-PSC measures its PSC value. This PSC value is defined as the average confidence generated over a range of parameter-scaled versions of the original model on the label, which is predicted by the original model. The larger the PSC value, the more likely the suspicious sample is poisoned. In particular, we start from the last layer of the deployed model and scale up different numbers of BN layers to obtain the scaled models. It is motivated by the previous findings (Huang et al., 2022; Jebreel et al., 2023) that trigger patterns often manifest as complicated features learned by the deeper layers of models, especially for those attacks with elaborate designs (Huang et al., 2022; Jebreel et al., 2023). To effectively determine the optimal number of layers for amplification, we design an adaptive algorithm by evaluating the scaling impact on the model's performance when processing benign samples.

In conclusion, our main contributions are four-fold. **(1)** We disclose an intriguing phenomenon, $i.e.$, parameter-oriented scaling consistency (PSC), where the prediction confidences of poisoned samples are more consistent than benign ones when scaling up BN parameters. **(2)** We provide theoretical insights to elucidate the PSC phenomenon. **(3)** We design a simple yet effective method (*i.e.*, IBD-PSC) to filter out poisoned testing images based on our findings. **(4)** We conduct extensive experiments on benchmark datasets, verifying the effectiveness of our method against 13 representative attacks and its resistance to potential adaptive attacks.

2. Related Work

2.1. Backdoor Attacks

In general, existing backdoor attacks can be categorized into three types based on the adversaries' capabilities: **(1)** poison-only attacks, **(2)** training-controlled attacks, and **(3)** model-controlled attacks. These attacks could happen whenever the training stage is not fully controlled.

Poison-only Backdoor Attacks. In these attacks, the adversaries can only manipulate the training dataset. Gu *et al.* (Gu et al., 2017) proposed the first poison-only attack (*i.e.*, BadNets). BadNets poisoned a few training samples by patching a predefined trigger, *e.g.*, a 3×3 white square, onto the bottom right corner of these samples. It then altered the labels of the modified samples to an adversaries-specified target label. Models trained on such poisoned training sets create a relation between the trigger and the target label. Subsequent studies further developed more stealthy attack methods, including invisible and clean-label attacks. The former methods (Chen et al., 2017; Li et al., 2021c) typically used imperceptible triggers to bypass manual detection, while the latter ones (Turner et al., 2019; Zeng et al., 2023; Gao et al., 2023) maintained the ground-truth label of poisoned samples. Besides, there are also the physical attack (Wenger et al., 2021; Gong et al., 2023; Xu et al., 2023) that adopt physical objects or spatial transformations as triggers and the adaptive attack methods (Tang et al., 2021; Qi et al., 2023) that are specifically designed to evade defenses.

Training-controlled Backdoor Attacks. In these attacks, adversaries can modify both the training dataset and the training process. One line of work aimed to circumvent existing defenses and human detection. For instance, the adversaries may introduce a 'noise mode' (Nguyen & Tran, 2020; 2021; Mo et al., 2024; Zhang et al., 2024) or incorpo-

rate well-designed regularization terms into training loss (Li et al., 2020; Doan et al., 2021; Xia et al., 2022). Another line of work focused on augmenting the effectiveness of attacks. For instance, Wang *et al.* (Wang et al., 2022c) exploited learning algorithms beyond supervised learning to ensure the correct injections of subtle triggers. Besides, Li *et al.* (Li et al., 2021c) and Zhang *et al.* (Zhang et al., 2022) introduced spatial transformations to poisoned samples to hide the triggers more robustly, extending the threat of backdoor attacks to the real physical scenarios.

Model-controlled Backdoor Attacks. In model-controlled backdoor attacks, adversaries modify model architectures or parameters directly to inject backdoors. For example, Tang *et al.* (Tang et al., 2020) implanted hidden backdoors by inserting an additional malicious module into the benign victim model. Qi *et al.* (Qi et al., 2022) proposed to maliciously modify the parameters of a narrow subnet in the benign model instead of inserting an additional module. This approach was more stealthy and was highly effective in both digital and physical scenarios.

Recently, a few works exploit backdoor attacks for positive purposes (Li et al., 2022a;c; 2023b; Guo et al., 2023c; Tang et al., 2023; Ya et al., 2024), which are out of our scope.

2.2. Backdoor Defenses

Based on the stage of the model lifecycle where defense occurs, existing defenses can be mainly divided into five main categories: **(1)** data purification (Tran et al., 2018; Li et al., 2021b; Jebreel et al., 2023), **(2)** poison suppression (Wang et al., 2022a; Huang et al., 2022; Tang et al., 2023), **(3)** model-level backdoor detection (Wang et al., 2019; Xiang et al., 2023; Wang et al., 2024; Yao et al., 2024; Wang et al., 2023; 2022b), **(4)** model-level backdoor mitigation (Liu et al., 2018; Zeng et al., 2022; Guo et al., 2023a; Li et al., 2024a;b; Xu et al., 2024), and **(5)** input-level backdoor detection (IBD) (Gao et al., 2021; Liu et al., 2023; Guo et al., 2023b). Specifically, data purification intends to filter out all poisoned samples in a given (third-party) dataset. It usually needs to train a model before identifying the influence of each training sample; Poison suppression aims to hinder the model's learning of the poisoned samples by modifying its training process to prevent backdoor creation; Model-level detection usually trains a meta-classifier or approximates trigger generation to determine whether a suspicious model contains hidden backdoors; IBD detects and prevents malicious inputs and acts as a 'firewall' of deployed models. In general, the first four strategies demand substantial computational resources since they typically necessitate model training or fine-tuning. However, these resources are unavailable for many researchers and developers, especially those using third-party models. This paper primarily focuses on IBD, which is more computation-friendly.

Previous IBD methods (Chou et al., 2020; Gao et al., 2021; Liu et al., 2023) are effective under implicit assumptions concerning the backdoor triggers. For example, STRIP (Gao et al., 2021) posited that trigger features play a dominant role, and the predictions of poisoned samples will not be affected even when benign features are overlaid. These assumptions can be easily circumvented by adaptive backdoor attacks (Nguyen & Tran, 2020; Li et al., 2021a; Duan et al., 2024). To the best of our knowledge, the most advanced IBD method is SCALE-UP (Guo et al., 2023b). It amplified all pixel values of an input sample with varying intensities and treated it as poisoned if the predictions were consistent. However, SCALE-UP inherited some potential limitations due to pixel value constraints (bounded in [0, 255]). For example, these constraints may alter predictions of poisoned samples, as amplification can transform higher pixel values into the maximum value of 255, causing triggers (*e.g.*, a white square) to disappear. How to design effective yet efficient IBD methods is still a critical open question.

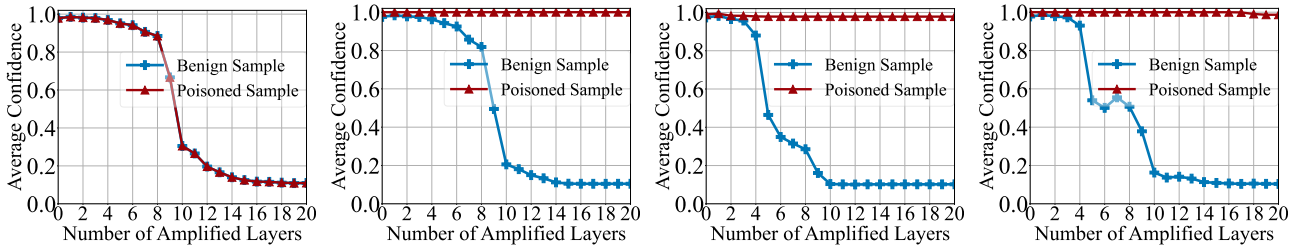
3. Parameter-oriented Scaling Consistency

As demonstrated in (Guo et al., 2023b), the predictions of poisoned samples are significantly more consistent than benign ones when amplifying all pixel values. Motivated by the fact that model predictions result from the co-effects of samples and model parameters, in this section, we explore whether a similar intriguing phenomenon still exists if we scale up model parameters instead of pixel values.

For simplicity, we mainly focus on the learnable parameters of BN layers since they are used to transform features and are widely exploited in almost all advanced DNNs. Before illustrating our key observation and its theoretical support, we first briefly review the mechanism of BN.

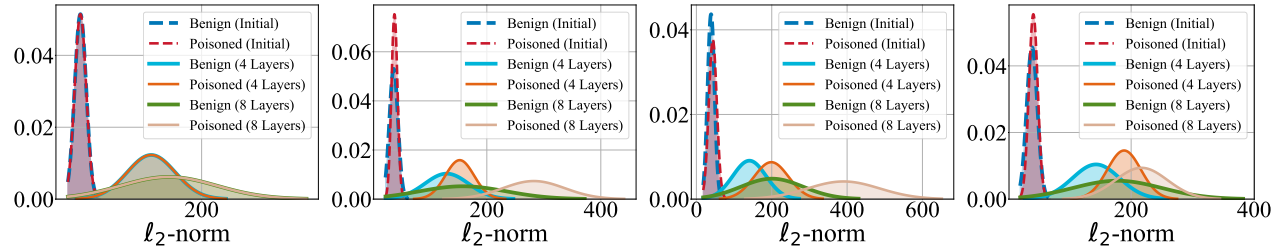
Batch Normalization. Let $\phi(\cdot;\gamma,\boldsymbol{\beta})$ denote the BN function, for a given batch feature maps $\boldsymbol{a}$, the BN operation transforms it into $\boldsymbol{b}$: $\boldsymbol{b}=\phi(\boldsymbol{a};\gamma,\boldsymbol{\beta})$. This transformation is expressed as $\phi(\boldsymbol{a};\gamma,\boldsymbol{\beta})=\gamma\left(\frac{\boldsymbol{a}-\boldsymbol{\mu}_a}{\sqrt{\sigma_a^2+\epsilon}}\right)+\boldsymbol{\beta}$, where ϵ is a small value, $\boldsymbol{\mu}_a$ and σ_a are the mean and standard deviation of $\boldsymbol{a}$, respectively. The γ and $\boldsymbol{\beta}$ are learnable parameters, designed to scale and shift normalized features, and learned during the training process.

Settings. In this section, we adopt BadNets (Gu et al., 2017), WaNet (Nguyen & Tran, 2021), and BATT (Xu et al., 2023) on CIFAR-10 (Krizhevsky et al., 2009) as examples for analyses. They are the representative of **(1)** patch-based attack, **(2)** sample-specific attack, and **(3)** physical attack, respectively. We exploit a standard ResNet-18 (He et al., 2016a) as our model structure. It contains twenty BN layers. For all attacks, we set the poisoning rate as 10%. Specifically, for each benign and poisoned image, we scale up on the BN



(a) Benign (b) BadNets (c) WaNet (d) BATT

Figure 2. The average confidence of benign and poisoned samples when amplifying different numbers of BN layers under benign and backdoored models (starting from the last layer).



(a) Benign Model (b) BadNets (c) WaNet (d) BATT

Figure 3. The approximated distribution of the ℓ_2-norm, fitted by Gaussian, of the final feature map of samples generated by models with different numbers of amplified BN layers. Increasing the number of amplified layers increases both value and variance of features.

parameters (*i.e.*, γ and $\boldsymbol{\beta}$) with $\omega = 1.5$ times starting from the last layer and gradually moving forward to more layers. Similar to (Guo et al., 2023b), we also calculate the *average confidence* defined as the average probability of samples on the label predicted by the original unamplified model. More details are in our Appendix B.

Results. As shown in Figure 2(a), the average prediction confidence of the poisoned and benign samples decreases at almost the same rate as the number of amplified BN layers increases under the benign model. In contrast, as shown in Figure 2(b)-Figure 2(d), the average prediction confidence of the poisoned samples remains nearly unchanged, whereas that of the benign samples also decreases during the parameter-amplified process under all three attacked models. In other words, benign and poisoned samples enjoy different BN-amplified prediction behaviors under attacked models. We call this intriguing phenomenon (of poisoned samples) as *parameter-oriented scaling consistency (PSC)*.

To verify that the PSC phenomenon is not accidental, we provide the following theoretical and empirical analyses.

Theorem 3.1. *Let $F = FC \circ f_L \circ \cdots \circ f_1$ be a backdoored DNN with L hidden layers and FC denotes the fully connected layers. Let $\boldsymbol{x}$ be an input, $\boldsymbol{b} = f_l \circ \cdots \circ f_1(\boldsymbol{x})$ be its batch-normalized feature after the l-th layer ($1 \leq l \leq L$), and t represent the attacker-specified target class. Assume that $\boldsymbol{b}$ follows a mixture of Gaussian distributions. Then the following two statements hold: (1) Amplifying the $\boldsymbol{\beta}$ and γ parameters of the l-th BN layer can make $\|\tilde{\boldsymbol{b}}\|_2$ ($\tilde{\boldsymbol{b}}$ is the amplified version of $\boldsymbol{b}$) arbitrarily large, and (2) There exists a positive constant M that is independent of $\tilde{\boldsymbol{b}}$, such that whenever $\|\tilde{\boldsymbol{b}}\|_2 > M$, then $\arg\max FC \circ f_L \circ \cdots \circ f_{l+1}(\tilde{\boldsymbol{b}}) = t$, even when $\arg\max FC \circ f_L \circ \cdots \circ f_{l+1}(\boldsymbol{b}) \neq t$.*

Theorem 3.1 indicates that larger enough feature norms can induce decreasing confidence in the original predicted class if the inputs are benign samples (under certain classical assumptions in learning theory). Poisoned samples, instead, will stay fine. Its proof is in Appendix A.

In practice, we find amplifying only a single BN layer may require an unreasonably large amplification factor and is unstable among different attacks or even BN layers, as demonstrated in Appendix C. Fortunately, as shown in Figure 3, amplifying multiple BN layers with a small factor (*e.g.*, 1.5) can also significantly increase the feature norm in the last pre-FC layer and is more stable across different settings. As such, we amplify multiple layers throughout this work.

4. The Proposed Method

4.1. Preliminaries

Threat Model. This work focuses on input-level backdoor detection under the white-box setting with limited computational capacities. Defenders have full access to the suspicious model downloaded from a third party, but they lack the resources to remove potential backdoors (via backdoor mitigation). Similar to prior works (Gao et al., 2021; Guo et al., 2023b), we assume that defenders have access to a limited number of local benign samples.

Defenders' Goals. An ideal IBD solution aims to precisely identify and eliminate all poisoned input samples while

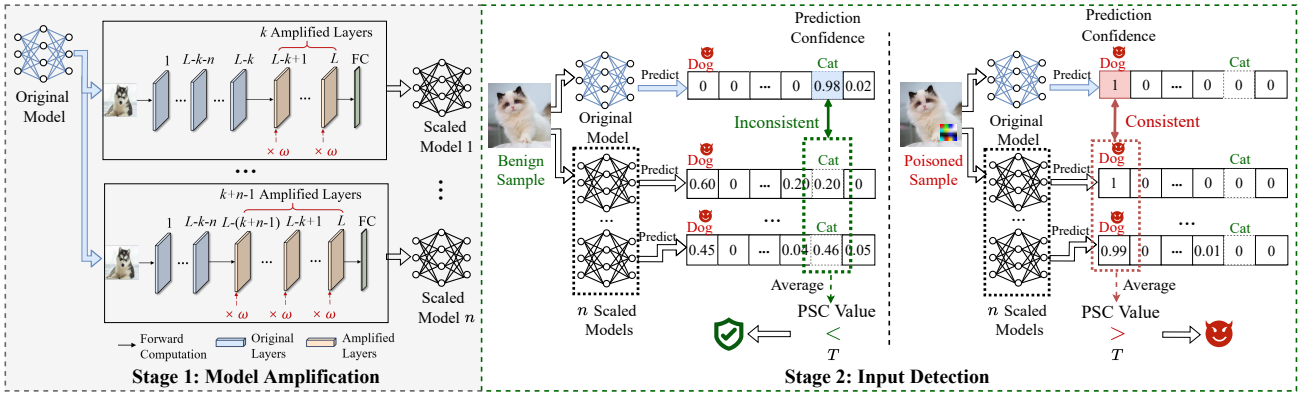


Figure 4. The main pipeline of our IBD-PSC. **Stage 1. Model Amplification**: Starting from the penultimate k-th layer of the original model, IBD-PSC gradually forward amplifies the parameters of more BN layers simultaneously to obtain n different parameter-amplified models. **Stage 2. Input Detection**: For each suspicious image, IBD-PSC will first calculate the prediction confidence of the obtained n parameter-amplified models on the label predicted by the original model. After that, IBD-PSC determines whether it is a poisoned sample by whether the average of obtained prediction confidences (defined as PSC value) is greater than a given threshold T.

preserving the inference efficiency of the deployed model. Consequently, defenders have two main goals: **(1)** Effectiveness: The defense should accurately identify whether a given suspicious image is malicious. **(2)** Efficiency: The defense must operate in real-time and integrate seamlessly as a plug-and-play module, ensuring minimal impact on the model's inference time.

The Overview of DNNs. Consider a DNN model $\mathcal{F}: \mathcal{X} \to [0,1]^C$ consisting L hidden layers, where $\mathcal{X}$ is the input space and C is the number of classes. We can specify it as

$$\mathcal{F} = \mathrm{FC} \circ f_L \circ f_{L-1} \circ \cdots \circ f_2 \circ f_1, \tag{1}$$

where FC denotes the fully-connected layers and f_i represents i-th hidden layer consisting of one convolutional, batch normalization, and activation layer.

The Main Pipeline of Backdoor Attack. Let $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ denote a training set, consisting of N *i.i.d.* samples. For each sample $(\boldsymbol{x}, y)$, $\boldsymbol{x} \in \mathcal{X} = [0,1]^{d_c \times d_w \times d_h}$ and $y \in \mathcal{Y} = \{1, 2, \cdots, C\}$, an adversary creates a poisoned training set $\hat{\mathcal{D}}$ by injecting a pre-defined trigger $\boldsymbol{t}$ into a subset of benign samples (*i.e.*, $\mathcal{D}_s$). The trigger is procured through a designated trigger generating function, symbolized as $\boldsymbol{t} = \tau(\boldsymbol{x})$, where $\tau: \mathcal{X} \to \mathcal{X}$. The generated poisoned samples are represented as $\mathcal{D}_p = \{(\hat{\boldsymbol{x}}, t) \mid \hat{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{t}, (\boldsymbol{x}, y) \in \mathcal{D}_s\}$. The final poisoned training set $\hat{\mathcal{D}}$ is formed by combining $\mathcal{D}_p$ with the remaining benign samples $\mathcal{D}_b$, *i.e.*, $\hat{\mathcal{D}} = \mathcal{D}_p \cup \mathcal{D}_b$. The poisoning rate is $\rho = |\mathcal{D}_p|/|\hat{\mathcal{D}}|$. The backdoor will be created for DNNs trained on the poisoned dataset $\hat{\mathcal{D}}$.

4.2. The Overview of IBD-PSC

As demonstrated in Section 3, the prediction confidences of poisoned samples exhibit greater consistency than those of benign ones when scaling up BN parameters of attacked DNNs. As such, we can detect whether a suspicious image is malicious by examining its parameter-oriented scaled consistency (PSC), a method we refer to as IBD-PSC.

In general, as shown in Figure 4, our IBD-PSC has two main stages, including **(1)** model amplification and **(2)** input detection. In the first stage, we amplify the BN parameters of different layers in the original model to obtain *a series of* parameter-amplified models. In the second stage, we calculate the PSC value of the suspicious image based on the obtained models and the original one. A larger PSC value indicates a higher likelihood that the suspicious image is malicious. The technical details are as follows.

4.3. Model Amplification

Overview. In this stage, we intend to obtain n different parameter-amplified versions of the original model by amplifying the parameters (*i.e.*, γ and $\boldsymbol{\beta}$) of its different BN layers. In particular, we amplify the later parts of the original model. It is motivated by the previous findings that trigger patterns often manifest as complicated features learned by the deeper (convolutional) layers of DNNs, especially for those attacks with elaborate designs (Huang et al., 2022; Jebreel et al., 2023). This finding is consistent with our observations in Figure 2. Specifically, let k denote the penultimate BN layers in which we scale up in the first parameter-amplified model. For the i-th amplified model, we scale up the parameters in the last $(k+i-1)$ BN layers with the same scaling factor ω. Let $\mathcal{F}$ denotes the original model, its parameter-amplified version containing k amplified BN layers with scaling factor ω (*i.e.*, $\hat{\mathcal{F}}_k^\omega$) can be defined as

$$\hat{\mathcal{F}}_k^\omega = \mathrm{FC} \circ \hat{f}_L^\omega \circ \hat{f}_{L-1}^\omega \circ \ldots \circ \hat{f}_{L-k+1}^\omega \circ \ldots \circ f_2 \circ f_1, \tag{2}$$

where $\hat{f}_i^\omega$ represents the BN layer of the i-th hidden layer undergoing an amplification process. It scales the original

Algorithm 1 Adaptive layer selection.

Input: original model $\mathcal{F}$, scaling factor ω, error rate threshold ξ, local benign dataset $\mathcal{D}_r$
Output: optimal number of amplified BN layers (*i.e.*, k) for the first parameter-amplified model
for $i \leftarrow 1$ **to** L **do**
  $k = i$
  Generate the parameter-amplified model $\hat{\mathcal{F}}_k^\omega$ using Equation (2)
  Calculate the error rate η using Equation (3)
  if $\eta > \xi$ **then**
    break
  end if
end for
return k

BN layer's parameters γ and $\boldsymbol{\beta}$ by a scaling factor ω, *i.e.*, $\hat{\gamma} = \omega \cdot \gamma$ and $\hat{\boldsymbol{\beta}} = \omega \cdot \boldsymbol{\beta}$. We also conduct ablation studies in Appendix D and Appendix E to assess the impact of amplifying BN layers in a forward sequential manner and that of amplifying all BN layers, respectively.

We exploit n instead of one parameter-amplified model (with many amplified BN layers) to balance the performance on benign and poisoned samples. In practice, n is a defender-assigned hyper-parameter. More details and its impact are included in Appendix M.3. Accordingly, the last remaining question for model amplification is selecting a suitable starting point k. Its technical details are as follows.

Layer Selection. To optimally determine the number of amplified BN layers, we design an adaptive algorithm to dynamically select a suitable k. Motivated by our PSC phenomenon (see Figure 2), we intend to *find the point where the prediction accuracy for benign samples begins to decline significantly*. Specifically, we incrementally increase k from 1 to L and monitor the error rate η. Let $\mathcal{D}_r$ denote the set of remaining benign samples. We can then compute the error rate η as the proportion of samples within $\mathcal{D}_r$ that are misclassified by the parameter-amplified model $\hat{\mathcal{F}}_k^\omega$, *i.e.*,

$$\eta = \frac{1}{|\mathcal{D}_r|} \sum_{(\boldsymbol{x},y)\in\mathcal{D}_r} \mathbb{I}\left(\operatorname{argmax}\left(\hat{\mathcal{F}}_k^\omega(\boldsymbol{x})\right) \neq y\right), \quad (3)$$

where $\mathbb{I}$ denotes the indicator function. Once η exceeds a predefined threshold ξ (*e.g.*, 60%), the BN layers from the $(L-k+1)$-th to the L-th layer are determined as the target layers for amplification. The details of the adaptive algorithm are outlined in Algorithm 1.

4.4. Input Detection

Once we obtain n parameter-amplified versions of the original model $\mathcal{F}$ with the starting amplified point k (*i.e.*, $\{\hat{\mathcal{F}}_{k+i-1}^\omega\}_{i=1}^n$), for each suspicious image, our IBD-PSC can examine it by calculating its PSC value based on their predictions. Specifically, we define the PSC value as the average confidence generated over parameter-amplified models on the label predicted by the original model, *i.e.*,

$$\text{PSC}(\boldsymbol{x}) = \frac{1}{n} \sum_{i=k}^{k+n-1} \hat{\mathcal{F}}_i^\omega(\boldsymbol{x})_{y'}, \quad (4)$$

where $y' = \operatorname{argmax}(\mathcal{F}(\boldsymbol{x}))$. After obtaining the PSC value, IBD-PSC assesses whether the input sample is malicious by comparing it to a predefined threshold T. If PSC $> T$, it is marked as a poisoned image.

5. Experiments

5.1. Experiment Settings

Datasets and Models. We follow the settings in existing backdoor defenses and conduct experiments on CIFAR-10 (Krizhevsky et al., 2009), GTSRB (Stallkamp et al., 2012) and a subset of ImageNet dataset with 200 classes (dubbed 'SubImageNet-200') (Deng et al., 2009) using the ResNet18 architecture (He et al., 2016a). More detailed settings are presented in Appendix F.

Attack Baselines. We evaluate the effectiveness of IBD-PSC against thirteen representative backdoor attacks, including **1)** BadNets (Gu et al., 2017), **2)** Blend (Chen et al., 2017), **3)** LC (Turner et al., 2019), **4)** ISSBA (Li et al., 2021a), **5)** TaCT (Tang et al., 2021), **6)** NARCISSUS (Zeng et al., 2023), **7)** Adap-Patch (Qi et al., 2023), **8)** BATT (Xu et al., 2023), **9)** PhysicalBA (Li et al., 2021c), **10)** IAD (Nguyen & Tran, 2020), **11)** WaNet (Nguyen & Tran, 2021), **12)** BPP (Wang et al., 2022c), and **13)** SRA (Qi et al., 2022). The first eight attacks are representative of poison-only attacks, while the last one is a model-controlled attack. The remaining four are training-controlled attacks. More details about the attack baselines are in the Appendix G.

Defense Settings. We compare our defense with classical and advanced input-level backdoor defenses, including STRIP (Gao et al., 2021), TeCo (Liu et al., 2023) and SCALE-UP (Guo et al., 2023b). We implement these defenses using their official codes with default settings. Our IBD-PSC defense maintains a consistent hyper-parameter setting across various attacks and datasets. Specifically, we set $\omega = 1.5$, $n = 5$, $\epsilon = 60\%$, and $T = 0.9$. Defenders can only access 100 benign samples as their local samples. More setting details about baseline methods are in Appendix H.

Evaluation Metrics. We employ two common metrics in our evaluation: **1)** the area under the receiver operating curve (AUROC) measures the overall performance of detection methods across different thresholds, and **2)** the F1 score measures both detection precision and recall.

Table 1. The performance (AUROC, F1) on the CIFAR-10 dataset. We mark the best result in boldface and failed cases (< 0.7) in red.

Attacks→	BadNets		Blend		PhysicalBA		IAD		WaNet		ISSBA		BATT		Avg.	
Defenses↓	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1
STRIP	0.931	0.842	0.453	0.114	0.884	0.882	0.962	0.907	0.469	0.125	0.364	0.526	0.449	0.258	0.663	0.494
TeCo	0.998	0.970	0.675	0.678	0.748	0.689	0.909	0.920	0.923	0.915	0.901	0.942	0.914	0.673	0.858	0.834
SCALE-UP	0.962	0.913	0.644	0.453	0.969	0.715	0.967	0.869	0.672	0.529	0.942	0.894	0.959	0.911	0.731	0.757
IBD-PSC	**1.000**	**0.967**	**0.998**	**0.960**	**0.972**	**0.942**	**0.983**	**0.952**	**0.984**	**0.956**	**1.000**	**0.986**	**0.999**	**0.966**	**0.992**	**0.961**

Table 2. The performance (AUROC, F1) on the GTSRB dataset. We mark the best result in boldface and failed cases (< 0.7) in red.

Attacks→	BadNets		Blend		PhysicalBA		IAD		WaNet		ISSBA		BATT		Avg.	
Defenses↓	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1
STRIP	0.962	0.915	0.426	0.088	0.700	0.479	0.855	0.890	0.356	0.201	0.640	0.625	0.648	0.368	0.657	0.588
TeCo	0.879	0.905	0.917	0.913	0.860	0.673	0.955	0.962	0.954	0.935	0.941	0.947	0.829	0.673	0.907	0.858
SCALE-UP	0.913	0.858	0.579	0.421	0.762	0.709	0.885	0.860	0.309	0.149	0.733	0.691	0.902	0.876	0.700	0.669
IBD-PSC	**0.968**	**0.965**	**0.953**	**0.928**	**0.940**	**0.946**	**0.970**	**0.971**	**0.986**	**0.973**	**0.972**	**0.971**	**0.969**	**0.968**	**0.969**	**0.962**

Table 3. The performance (AUROC, F1) on SubImageNet-200. We mark the best result in boldface and failed cases (< 0.7) in red.

Attacks→	BadNets		Blend		PhysicalBA		IAD		WaNet		ISSBA		BATT		Avg.	
Defenses↓	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1
STRIP	0.840	0.828	0.799	0.772	0.618	0.468	0.528	0.419	0.563	0.356	0.768	0.765	0.554	0.361	0.681	0.596
TeCo	0.978	0.880	0.958	0.849	0.926	0.842	0.927	0.920	0.903	0.747	0.945	0.921	0.690	0.692	0.908	0.846
SCALE-UP	0.967	0.895	0.531	0.356	0.932	0.876	0.322	0.030	0.563	0.356	0.945	0.912	0.967	0.921	0.725	0.651
IBD-PSC	**1.000**	**0.992**	**0.989**	**0.833**	**0.994**	**0.988**	**0.994**	**0.996**	**0.967**	**0.981**	**0.989**	**0.987**	**0.998**	**0.998**	**0.990**	**0.974**

5.2. Main Results

As shown in Table 1-3, IBD-PSC consistently achieves promising performance in all cases across various datasets. For instance, it achieves AUROC and F1 scores approaching 1.0, indicating its effectiveness in various attack scenarios. The results also demonstrate that IBD-PSC achieves a substantial improvement in detection performance compared to the defense baselines. In contrast, all baseline defenses fail in some cases (marked in red), especially under attacks involving subtle alterations across multiple pixels (*e.g.*, Blend, WaNet) or physical attacks. This failure is primarily caused by their implicit assumptions about backdoors, such as sample-agnostic triggers and robustness against image preprocessing. We also provide the results with PreActResNet18 (He et al., 2016b) and MobileNet (Krizhevsky et al., 2009) architectures in Appendix I. We also provide the ROC curves of defenses against four representative attacks in Appendix J. Besides, for more experimental results under other attack baselines in Appendix K.

We also calculated the inference time of all methods under identical and ideal conditions for evaluating efficiency. For example, we assume that defenders will load all required models and images simultaneously (with more memory requirements compared to the vanilla model inference). Arguably, this comparison is fair and reasonable since different defenses differ greatly in their mechanisms and requirements. Detailed settings can be found in Appendix L. As shown in Figure 5, the efficiency of our IBD-PSC is on par with or even better than all baseline defenses. The extra time is negligible compared to no defense, although IBD-PSC may increase some storage or computational consumptions. More detailed discussions about our running efficiency and storage requirements are in Appendix R.

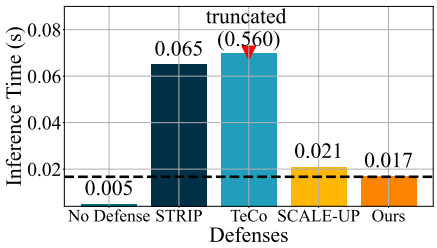
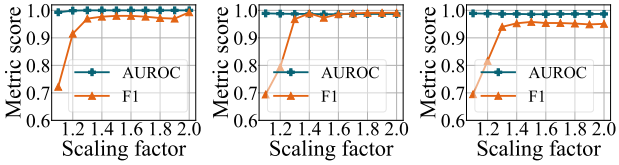


Figure 5. The inference time on the CIFAR-10 dataset.

5.3. Ablation Study

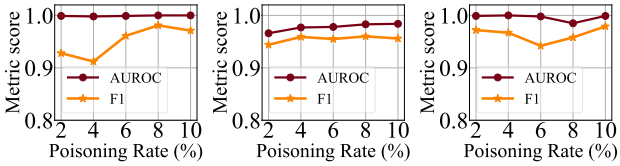
Impact of Scaling Factor ω. IBD-PSC generates scaled models by amplifying the learnable parameter values of the selected BN layers with a fixed scalar ω. We hereby explore its effects on our method. Specifically, we vary ω from 1 to 2 and calculate the AUROC and F1 scores of IBD-PSC against three representative backdoor attacks (*i.e.*, BadNets, WaNet, and BATT) on CIFAR-10. As shown in Figure 6, in the initial phase, increasing ω can significantly improve both the AUROC and F1 scores against different backdoor attacks. Furthermore, the AUROC and F1 scores converge to nearly one and stabilize at approximately one for ω values of 1.5 or higher, *i.e.*, the scaling factor has a relatively minor influence when it is sufficiently large. Besides, we conduct further ablation studies on other hyper-parameters of our method, as detailed in Appendix M.

Impact of Confidence Consistency. IBD-PSC leverages



(a) BadNets (b) WaNet (c) BATT

Figure 6. The impact of scaling factors on CIFAR-10.



(a) BadNets (b) WaNet (c) BATT

Figure 7. The impact of poisoning rate on CIFAR-10.

the consistency of confidence for detection, in contrast to the SCALE-UP method, which relies on the consistency of the predicted label. SCALE-UP is designed for black-box scenarios where defenders only have access to predicted labels, while our IBD-PSC focuses on white-box settings where predicted confidences are naturally available. To validate the effectiveness of IBD-PSC, we develop a variant that uses label consistency (dubbed 'Ours-L'). We then calculate the False Positive Rate (FPR) (%) for both target and benign classes on the CIFAR-10 dataset across various backdoor attacks. As shown in Table 4, our method significantly reduces false positives in both the target and benign classes, outperforming both the Ours-L and SCALE-UP.

5.4. Resistance to Potential Adaptive Attacks

We initially assess the performance of IBD-PSC against attacks with low poisoning rates. This is because a small poisoning rate ρ can prevent models from over-fitting triggers, thus weakening the association between triggers and target labels, as demonstrated in previous studies (Guo et al., 2023b; Qi et al., 2023). Specifically, we conduct attacks (BadNets, WaNet, and BATT) on the CIFAR-10 dataset with ρ ranging from 0.02 to 0.1, ensuring the attack success rates exceed 80%. The results in Figure 7 consistently demonstrate the effectiveness of IBD-PSC, with AUROC and F1 scores consistently above 0.98 and 0.95, respectively. Results on SubImageNet-200 are shown in Appendix N.1.

We further evaluate the robustness of IBD-PSC against potential adaptive attacks in the worst-case scenario, where adversaries possess complete knowledge of our defense. Typically, a vanilla backdoored model functions normally with benign samples but yields adversary-specific predictions when exposed to poisoned samples. The loss function for training such backdoored models is defined as follows:

$$\mathcal{L}_{\text{bd}} = \sum_{i=1}^{|\mathcal{D}_b|} \mathcal{L}(\mathcal{F}(\boldsymbol{x}_i), y_i) + \sum_{j=1}^{|\mathcal{D}_p|} \mathcal{L}(\mathcal{F}(\boldsymbol{x}_j), y_t), \quad (5)$$

Table 4. The False Positive Rate (FPR) (%) of SCALE-UP and our defense on target and benign classes on the CIFAR-10 dataset.

Defenses→	SCALE-UP		Ours-L		Ours	
AttackS↓	Target	Benign	Target	Benign	Target	Benign
BadNets	72.74	29.00	0.40	9.76	0.20	1.88
Blend	54.28	19.80	22.55	3.39	18.34	2.64
PhysicalBA	90.58	23.98	4.60	5.42	4.10	1.50
WaNet	76.70	28.11	81.41	10.05	69.20	8.16
ISSBA	93.93	20.70	20.94	3.00	17.22	0.61
BATT	57.74	18.78	2.35	9.72	0.87	6.90
SRA	65.55	29.33	0.62	10.48	0.50	10.13
Ada-Patch	93.80	25.77	8.67	4.78	4.34	3.00

Table 5. Performance of IBD-PSC under adaptive attacks.

α→	0.2		0.5		0.9		0.99	
Attacks↓	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1
BadNets	0.992	0.978	0.986	0.964	0.995	0.962	0.996	0.951
WaNet	0.947	0.949	0.956	0.942	0.931	0.927	0.819	0.862
BATT	0.986	0.968	0.994	0.956	0.982	0.975	0.979	0.959

where $\mathcal{L}(\cdot)$ represents the cross entropy loss function.

We design an adaptive loss term $\mathcal{L}_{\text{ada}}$ to ensure benign samples are correctly predicted under parameter amplification:

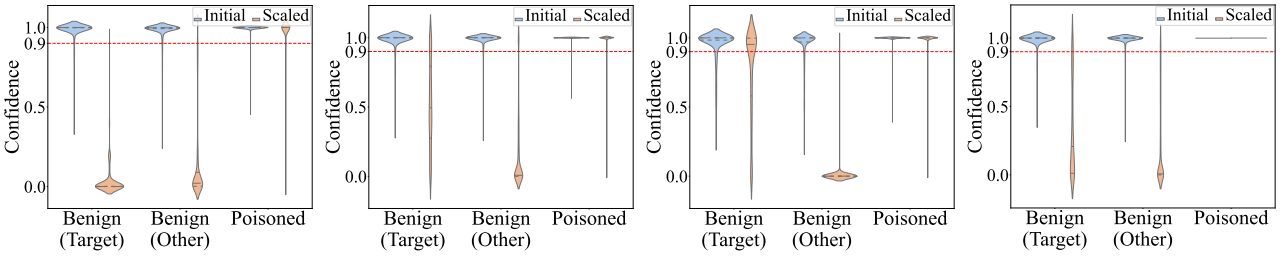
$$\mathcal{L}_{\text{ada}} = \sum_{i=1}^{|\mathcal{D}_b|} \mathcal{L}(\hat{\mathcal{F}}_k^{\omega}(\boldsymbol{x}_i; \hat{\boldsymbol{\theta}}), y_i). \quad (6)$$

Subsequently, we integrate this adaptive loss $\mathcal{L}_{\text{ada}}$ with the vanilla loss $\mathcal{L}_{\text{bd}}$ to formulate the overall loss function as $\mathcal{L} = \alpha\mathcal{L}_{\text{bd}} + (1-\alpha)\mathcal{L}_{\text{ada}}$, where α is a weighting factor. We then optimize the original model's parameters $\boldsymbol{\theta}$ by minimizing $\mathcal{L}$ during the training phase.

Similar to previous experiments, we also employ the three representative backdoor attacks to develop adaptive attacks on the CIFAR-10 dataset. Table 5 demonstrates the sustained robustness of our IBD-PSC across all cases. The effectiveness primarily originates from our adaptive layer selection strategy, which dynamically identifies BN layers for amplification, regardless of whether it is a vanilla or an adaptive backdoored model. The layers selected during the inference stage typically differ from those used in the training phase, enabling the IBD-PSC to effectively detect poisoned samples. More results and the resistance to another adaptive attack can be found in Appendix N.2.

5.5. Performance on Benign Samples from Target Class

In this section, we evaluate the effectiveness of our defense on benign samples from the target class. We conduct experiments on the CIFAR-10 dataset against four different attacks and present the confidences of both the initial backdoored model and the scaled models in Figure 8. As shown, the confidences of benign samples from both the target class and other classes decrease due to parameter amplification, falling below the threshold. In contrast, the confidence values for poisoned samples mostly remain above this thresh-



(a) BadNets (b) Blend (b) BATT (f) Ada-patch

Figure 8. The violin plots of the prediction confidences for benign samples in the target and other classes, as well as for poisoned samples, as predicted by the initial and scaled models on CIFAR-10. The threshold is 0.9.

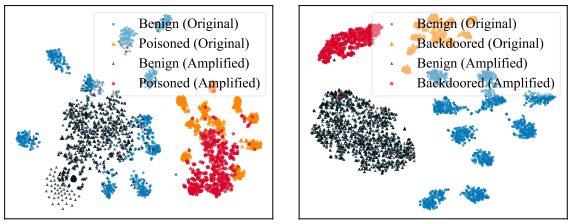
old. These results demonstrate that our defense effectively distinguishes between benign and poisoned samples, regardless of whether the benign sample originates from the target class. In particular, we observe an interesting phenomenon that scaled models tend to cluster the confidences for benign samples from the target class in the more difficult-to-learn class(es), rather than in the easier ones, which is unexpected. We will further explore its intrinsic mechanism in our future work. Additional analysis can be found in Appendix O.

5.6. A Closer Look to the Effectiveness of our Method

To gain deeper insights, we delve into the mechanisms of both SCALE-UP and our IBD-PSC. We utilize t-SNE (Van der Maaten & Hinton, 2008) for visualizing the features of benign and poisoned samples in the last hidden layer. We adopt the representative BadNets attack method on the CIFAR-10 dataset as an example for our discussions. More results about other attack methods can be found in Appendix P. The results in Figure 9 demonstrate that both SCALE-UP and our IBD-PSC induce more significant shifts in the feature space for benign samples compared to the poisoned samples. These larger shifts result in changes in the predictions for benign samples. These results provide clear evidence of the effectiveness of the two defense methods. Furthermore, in contrast to SCALE-UP, our IBD-PSC method induces more significant shifts in benign samples. This disparity in shift magnitude may stem from the constrained pixel value range of [0, 255], potentially mitigating the impact of amplification. However, the values of model parameters do not have such bounded constraints. Consequently, the larger shifts contribute to a more distinct separation between benign and poisoned samples, significantly augmenting the effectiveness of IBD-PSC.

5.7. The Extension to Training Set Purification

Although our method is initially and primarily designed to filter malicious testing samples, it can also be used to detect potentially poisoned samples within a compromised training set. Specifically, users can first train a model on this dataset with a standard process and then exploit our detection method. To verify our effectiveness, we conduct experiments on the CIFAR-10 dataset against three representative attacks. The results show a 100% TPR and nearly 100% AUROC scores, with FPR scores close to 0%. We compare the detection performance of our method with the most advanced defenses, *i.e.*, CD (Huang et al., 2023) and MSPC (Pal et al., 2024), and the results show that our method achieves the best detection performance. Subsequently, we retrain a model on this purified dataset to evaluate both its BA and the ASR. The ASR scores of these retrained models are less than 0.5%, rendering the attacks ineffective. More settings and results can be found in Appendix Q.



(a) SCALE-UP (b) Ours

Figure 9. t-SNE of feature representations of benign and poisoned samples on the CIFAR-10 dataset against BadNets attack.

6. Conclusion

In this paper, we proposed a simple yet effective method (dubbed IBD-PSC) for determining whether a suspicious image is poisoned. The IBD-PSC was inspired by our discovery of an intriguing phenomenon, named parameter-oriented scaled consistency (PSC). This phenomenon manifests through a significant uniformity of prediction confidences for poisoned samples, in contrast to benign ones, when the parameters of selected BN layers undergo amplification. We provided the theoretical and empirical foundations to support this phenomenon. To enhance the detection performance, we also designed an adaptive algorithm to dynamically select the number of BN layers for amplification. We conducted thirteen backdoor attack methods on benchmark datasets to comprehensively verify the effectiveness of our IBD-PSC. We also demonstrated that our IBD-PSC is highly efficient and resistant to potential adaptive attacks.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grants 62071142 and by the Guangdong Basic and Applied Basic Research Foundation under Grant 2024A1515012299.

Impact Statement

Backdoor attacks have posed severe threats in DNNs since developers often rely on external untrustworthy training resources (*e.g.*, datasets and model backbones). This paper proposes a simple yet effective input-level backdoor detection to identify and filter malicious testing samples. It generally has no ethical issues since it does not expose new vulnerabilities within DNNs and is purely defensive. However, we need to notice that our work can only filter out poisoned input images but cannot repair potential backdoors in the deployed model. Besides, it cannot recover trigger patterns or the ground-truth class of the poisoned samples. People should not be too optimistic about eliminating backdoor threats. Moreover, the adversaries may design more advanced backdoor attacks against our defense, although we have demonstrated that it is challenging. People should use only trusted training resources and models to eliminate and prevent backdoor attacks at the source.

References

Bai, J., Wu, B., Zhang, Y., Li, Y., Li, Z., and Xia, S.-T. Targeted Attack against Deep Neural Networks via Flipping Limited Weight Bits. In *ICLR*, 2021.

Chen, B., Carvalho, W., Baracaldo, N., Ludwig, H., Edwards, B., Lee, T., Molloy, I., and Srivastava, B. Detecting backdoor attacks on deep neural networks by activation clustering. In *CEUR Workshop*, 2018.

Chen, W., Wu, B., and Wang, H. Effective backdoor defense by exploiting sensitivity of poisoned samples. In *NeurIPS*, 2022.

Chen, X., Liu, C., Li, B., Lu, K., and Song, D. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning. *arXiv*, 2017.

Chou, E., Tramèr, F., Pellegrino, G., and Boneh, D. SentiNet: Detecting physical attacks against deep learning systems. In *IEEE S&P Workshop*, 2020.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009.

Doan, K., Lao, Y., and Li, P. Backdoor Attack with Imperceptible Input and Latent Modification. In *NeurIPS*, 2021.

Duan, Q., Hua, Z., Liao, Q., Zhang, Y., and Zhang, L. Y. Conditional backdoor attack via jpeg compression. In *AAAI*, 2024.

Gao, Y., Kim, Y., Doan, B. G., Zhang, Z., Zhang, G., Nepal, S., Ranasinghe, D. C., and Kim, H. Design and Evaluation of a Multi-Domain Trojan Detection Method on Deep Neural Networks. *IEEE Transactions on Dependable and Secure Computing*, 2021.

Gao, Y., Li, Y., Zhu, L., Wu, D., Jiang, Y., and Xia, S.-T. Not all samples are born equal: Towards effective clean-label backdoor attacks. *PR*, 2023.

Gong, X., Wang, Z., Chen, Y., Xue, M., Wang, Q., and Shen, C. Kaleidoscope: Physical Backdoor Attacks against Deep Neural Networks with RGB Filters. *IEEE Transactions on Dependable and Secure Computing*, 2023.

Gu, T., Dolan-Gavitt, B., and Garg, S. BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. *IEEE Access*, 2017.

Guo, H., Lu, C., Bao, F., Pang, T., Yan, S., Du, C., and Li, C. Gaussian mixture solvers for diffusion models. In *NeurIPS*, 2024.

Guo, J., Li, A., Wang, L., and Liu, C. PolicyCleanse: Backdoor Detection and Mitigation in Reinforcement Learning. In *ICCV*, 2023a.

Guo, J., Li, Y., Chen, X., Guo, H., Sun, L., and Liu, C. SCALE-UP: An efficient black-box input-level backdoor detection via analyzing scaled prediction consistency. In *ICLR*, 2023b.

Guo, J., Li, Y., Wang, L., Xia, S.-T., Huang, H., Liu, C., and Li, B. Domain watermark: Effective and harmless dataset copyright protection is closed at hand. In *NeurIPS*, 2023c.

Hayase, J., Kong, W., Somani, R., and Oh, S. Spectre: Defending against backdoor attacks using robust statistics. In *ICML*, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *CVPR*, 2016a.

He, K., Zhang, X., Ren, S., and Sun, J. Identity Mappings in Deep Residual Networks. In *ICCV*, 2016b.

Huang, H., Ma, X., Erfani, S., and Bailey, J. Distilling cognitive backdoor patterns within an image. In *ICLR*, 2023.

Huang, K., Li, Y., Wu, B., Qin, Z., and Ren, K. Backdoor Defense via Decoupling the Training Process. In *ICLR*, 2022.

Jebreel, N. M., Domingo-Ferrer, J., and Li, Y. Defending Against Backdoor Attacks by Layer-wise Feature Analysis. In *SIGKDD*, 2023.

Krizhevsky, A., Hinton, G., et al. Learning Multiple Layers of Features from Tiny Images. *Technical report*, 2009.

Li, B., Cai, Y., Cai, J., Li, Y., Qiu, H., Wang, R., and Zhang, T. Purifying quantization-conditioned backdoors via layer-wise activation correction with distribution approximation. In *ICML*, 2024a.

Li, B., Cai, Y., Li, H., Xue, F., Li, Z., and Li, Y. Nearest is not dearest: Towards practical defense against quantization-conditioned backdoor attacks. In *CVPR*, 2024b.

Li, S., Xue, M., Zhao, B. Z. H., Zhu, H., and Zhang, X. Invisible Backdoor Attacks on Deep Neural Networks via Steganography and Regularization. *IEEE Transactions on Dependable and Secure Computing*, 2020.

Li, Y., Li, Y., Wu, B., Li, L., He, R., and Lyu, S. Invisible Backdoor Attack with Sample-Specific Triggers. In *ICCV*, 2021a.

Li, Y., Lyu, X., Koren, N., Lyu, L., Li, B., and Ma, X. Anti-Backdoor Learning: Training Clean Models on Poisoned Data. In *NeurIPS*, 2021b.

Li, Y., Zhai, T., Jiang, Y., Li, Z., and Xia, S.-T. Backdoor Attack in the Physical World. In *ICLR Workshop*, 2021c.

Li, Y., Bai, Y., Jiang, Y., Yang, Y., Xia, S.-T., and Li, B. Untargeted backdoor watermark: Towards harmless and stealthy dataset copyright protection. In *NeurIPS*, 2022a.

Li, Y., Jiang, Y., Li, Z., and Xia, S.-T. Backdoor Learning: A Survey. *IEEE Transactions on Neural Networks and learning systems*, 2022b.

Li, Y., Zhu, L., Jia, X., Jiang, Y., Xia, S.-T., and Cao, X. Defending against model stealing via verifying embedded external features. In *AAAI*, 2022c.

Li, Y., Mengxi, Y., Yang, B., Yong, J., and Shu-Tao, X. BackdoorBox: A Python Toolbox for Backdoor Learning. In *ICLR Workshop*, 2023a.

Li, Y., Zhu, M., Yang, X., Jiang, Y., Wei, T., and Xia, S.-T. Black-box dataset ownership verification via backdoor watermarking. *IEEE Transactions on Information Forensics and Security*, 2023b.

Liu, K., Dolan-Gavitt, B., and Garg, S. Fine-Pruning: Defending Against Backdooring Attacks on Deep Neural Networks. In *RAID*, 2018.

Liu, X., Li, M., Wang, H., Hu, S., Ye, D., Jin, H., Wu, L., and Xiao, C. Detecting Backdoors During the Inference Stage Based on Corruption Robustness Consistency. In *CVPR*, 2023.

Loureiro, B., Sicuro, G., Gerbelot, C., Pacco, A., Krzakala, F., and Zdeborová, L. Learning gaussian mixtures with generalized linear models: Precise asymptotics in high-dimensions. In *NeurIPS*, 2021.

Ma, W., Wang, D., Sun, R., Xue, M., Wen, S., and Xiang, Y. The" beatrix"resurrections: Robust backdoor detection via gram matrices. In *NDSS*, 2022.

Mo, X., Zhang, Y., Zhang, L. Y., Luo, W., Sun, N., Hu, S., Gao, S., and Xiang, Y. Robust backdoor detection for deep learning via topological evolution dynamics. *IEEE S&P*, 2024.

Nguyen, T. A. and Tran, A. Input-Aware Dynamic Backdoor Attack. In *NeurIPS*, 2020.

Nguyen, T. A. and Tran, A. T. WaNet – Imperceptible Warping-based Backdoor Attack. In *ICLR*, 2021.

Pal, S., Yao, Y., Wang, R., Shen, B., and Liu, S. Backdoor secrets unveiled: Identifying backdoor data with optimized scaled prediction consistency. In *ICLR*, 2024.

Pan, M., Zeng, Y., Lyu, L., Lin, X., and Jia, R. {ASSET}: Robust backdoor data detection across a multiplicity of deep learning paradigms. In *USENIX Security*, 2023.

Papyan, V., Han, X., and Donoho, D. L. Prevalence of neural collapse during the terminal phase of deep learning training. *PNAS*, 2020.

Peri, N., Gupta, N., Huang, W. R., Fowl, L., Zhu, C., Feizi, S., Goldstein, T., and Dickerson, J. P. Deep k-nn defense against clean-label data poisoning attacks. In *ECCV*, 2020.

Qi, X., Xie, T., Pan, R., Zhu, J., Yang, Y., and Bu, K. Towards Practical Deployment-Stage Backdoor Attack on Deep Neural Networks. In *CVPR*, 2022.

Qi, X., Xie, T., Li, Y., Mahloujifar, S., and Mittal, P. Revisiting the Assumption of Latent Separability for Backdoor Defenses. In *ICLR*, 2023.

Stallkamp, J., Schlipsing, M., Salmen, J., and Igel, C. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks*, 2012.

Tang, D., Wang, X., Tang, H., and Zhang, K. Demon in the Variant: Statistical Analysis of DNNs for Robust Backdoor Contamination Detection. In *USENIX Security*, 2021.

Tang, R., Du, M., Liu, N., Yang, F., and Hu, X. An Embarrassingly Simple Approach for Trojan Attack in Deep Neural Networks. In *SIGKDD*, 2020.

Tang, R., Yuan, J., Li, Y., Liu, Z., Chen, R., and Hu, X. Setting the Trap: Capturing and Defeating Backdoor Threats in PLMs through Honeypots. In *NeurIPS*, 2023.

Tishby, N. and Zaslavsky, N. Deep Learning and the Information Bottleneck Principle. In *ITW*, 2015.

Tran, B., Li, J., and Madry, A. Spectral Signatures in Backdoor Attacks. In *NeurIPS*, 2018.

Turner, A., Tsipras, D., and Madry, A. Label-Consistent Backdoor Attacks. *arXiv*, 2019.

Van der Maaten, L. and Hinton, G. Visualizing data using t-SNE. *JMLR*, 2008.

Wang, B., Yao, Y., Shan, S., Li, H., Viswanath, B., Zheng, H., and Zhao, B. Y. Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks. In *IEEE S&P*, 2019.

Wang, H., Xiang, Z., Miller, D. J., and Kesidis, G. MM-BD: Post-Training Detection of Backdoor Attacks with Arbitrary Backdoor Pattern Types Using a Maximum Margin Statistic. In *IEEE S&P*, 2024.

Wang, Z., Ding, H., Zhai, J., and Ma, S. Training with More Confidence: Mitigating Injected and Natural Backdoors During Training. In *NeurIPS*, 2022a.

Wang, Z., Mei, K., Ding, H., Zhai, J., and Ma, S. Rethinking the reverse-engineering of trojan triggers. In *NeurIPS*, 2022b.

Wang, Z., Zhai, J., and Ma, S. BppAttack: Stealthy and Efficient Trojan Attacks against Deep Neural Networks via Image Quantization and Contrastive Adversarial Learning. In *CVPR*, 2022c.

Wang, Z., Mei, K., Zhai, J., and Ma, S. Unicorn: A unified backdoor trigger inversion framework. In *ICLR*, 2023.

Wenger, E., Passananti, J., Bhagoji, A. N., Yao, Y., Zheng, H., and Zhao, B. Y. Backdoor Attacks Against Deep Learning Systems in the Physical World. In *CVPR*, 2021.

Xia, P., Niu, H., Li, Z., and Li, B. Enhancing backdoor attacks with multi-level mmd regularization. *IEEE Transactions on Dependable and Secure Computing*, 2022.

Xiang, Z., Xiong, Z., and Li, B. Umd: Unsupervised model detection for x2x backdoor attacks. In *ICML*, 2023.

Xu, T., Li, Y., Jiang, Y., and Xia, S.-T. Batt: Backdoor attack with transformation-based triggers. In *ICASSP*, 2023.

Xu, X., Huang, K., Li, Y., Qin, Z., and Ren, K. Towards reliable and efficient backdoor trigger inversion via decoupling benign features. In *ICLR*, 2024.

Ya, M., Li, Y., Dai, T., Wang, B., Jiang, Y., and Xia, S.-T. Towards faithful xai evaluation via generalization-limited backdoor watermark. In *ICLR*, 2024.

Yao, Z., Zhang, H., Guo, Y., Tian, X., Peng, W., Zou, Y., Zhang, L. Y., and Chen, C. Reverse backdoor distillation: Towards online backdoor attack detection for deep neural network models. *IEEE Transactions on Dependable and Secure Computing*, 2024.

Zeng, Y., Park, W., Mao, Z. M., and Jia, R. Rethinking the backdoor attacks' triggers: A frequency perspective. In *ICCV*, 2021.

Zeng, Y., Chen, S., Park, W., Mao, Z. M., Jin, M., and Jia, R. Adversarial Unlearning of Backdoors via Implicit Hypergradient. In *ICLR*, 2022.

Zeng, Y., Pan, M., Just, H. A., Lyu, L., Qiu, M., and Jia, R. Narcissus: A Practical Clean-Label Backdoor Attack with Limited Information. In *CCS*, 2023.

Zhang, H., Hu, S., Wang, Y., Zhang, L. Y., Zhou, Z., Wang, X., Zhang, Y., and Chen, C. Detector collapse: Backdooring object detection to catastrophic overload or blindness. In *IJCAI*, 2024.

Zhang, J., Dongdong, C., Huang, Q., Liao, J., Zhang, W., Feng, H., Hua, G., and Yu, N. Poison ink: Robust and invisible backdoor attack. *IEEE Transactions on Image Processing*, 2022.

Zoran, D. and Weiss, Y. Natural images, gaussian mixtures and dead leaves. In *NeurIPS*, 2012.

Appendix

A. The Omitted Proof of Theorem 3.1

Theorem 3.1. *Let $F = FC \circ f_L \circ \cdots \circ f_1$ be a backdoored DNN with L hidden layers and FC denotes the fully-connected layers. Let x be an input, $\boldsymbol{b} = f_l \circ \cdots \circ f_1(x)$ be its batch-normalized feature after the l-th layer ($1 \leq l \leq L$), and t represent the attacker-specified target class. Assume that $\boldsymbol{b}$ follows a mixture of Gaussian distribution. Then the following two statements hold: (1) Amplifying the $\boldsymbol{\beta}$ and γ parameters of the l-th BN layer can make $\|\tilde{\boldsymbol{b}}\|_2$ ($\tilde{\boldsymbol{b}}$ is the amplified version of $\boldsymbol{b}$) arbitrarily large, and (2) There exists a positive constant M that is independent of $\tilde{\boldsymbol{b}}$, such that whenever $\|\tilde{\boldsymbol{b}}\|_2 > M$, then $\arg\max FC \circ f_L \circ \cdots \circ f_{l+1}(\tilde{\boldsymbol{b}}) = t$, even when $\arg\max FC \circ f_L \circ \cdots \circ f_{l+1}(\boldsymbol{b}) \neq t$*

Proof of Theorem 3.1: For simplicity, let $\mathcal{F}$ denote the benign model and $\tilde{\mathcal{F}}$ denote the backdoored model. We look at the l-th (pre-batch-norm) feature layer $\boldsymbol{a}_l$ such that

$$\boldsymbol{a} = \boldsymbol{a}_l(\boldsymbol{x}), \boldsymbol{b} = \phi(\boldsymbol{a}; \gamma_{\text{Benign}}, \boldsymbol{\beta}_{\text{Benign}}), \mathcal{F}(\boldsymbol{x}) = \text{FC} \circ f_L \circ \cdots \circ f_{l+1}(\boldsymbol{b}), \tag{A1}$$

$$\tilde{\boldsymbol{a}} = \tilde{\boldsymbol{a}}_l(\boldsymbol{x}), \tilde{\boldsymbol{b}} = \phi(\tilde{\boldsymbol{a}}; \gamma, \boldsymbol{\beta}), \tilde{\mathcal{F}}(\boldsymbol{x}) = \text{FC} \circ f_L \circ \cdots \circ f_{l+1}(\tilde{\boldsymbol{b}}). \tag{A2}$$

We assume all features follow the mixture of Gaussians, an assumption commonly used in many deep learning theory papers (Guo et al., 2024; Zoran & Weiss, 2012; Loureiro et al., 2021) as it simplifies analysis and provides a tractable framework for modeling complex data distributions. Consequently, $\boldsymbol{a}$ and $\tilde{\boldsymbol{a}}$ follow:

$$\boldsymbol{a} \sim \frac{1}{C}\sum_{c=1}^{C} Z_c \exp\frac{-\|\boldsymbol{a}-\mu_c\|_2^2}{2\sigma_c^2}, \boldsymbol{a}^c|c \sim Z_c \exp\frac{-\|\boldsymbol{a}-\mu_c\|_2^2}{2\sigma_c^2}, c \in \mathcal{Y}, \tag{A3}$$

$$\boldsymbol{b}^c|c \sim B_c \exp -\frac{\|\boldsymbol{b}^c - (\boldsymbol{\beta}_{\text{Benign}} - \gamma_{\text{Benign}}\frac{\mu_{\boldsymbol{a}}-\mu_c}{\sqrt{\sigma_{\boldsymbol{a}}^2+\epsilon}})\|_2^2}{2\frac{\gamma^2\sigma_c^2}{\sigma_{\boldsymbol{a}}^2}} = \mathcal{N}(\boldsymbol{a}; \boldsymbol{\beta}_{\text{Benign}} - \gamma_{\text{Benign}}\frac{\mu_{\boldsymbol{a}}-\mu_c}{\sqrt{\sigma_{\boldsymbol{a}}^2+\epsilon}}, \frac{\gamma_{\text{Benign}}\sigma_c}{\sigma_{\boldsymbol{a}}}), c \in \mathcal{Y}, \tag{A4}$$

and

$$\tilde{\boldsymbol{a}} \sim \frac{1}{C}\sum_{c=1}^{C} Z_c \exp\frac{-\|\tilde{\boldsymbol{a}}-\tilde{\mu}_c\|_2^2}{2\tilde{\sigma}_c^2}, \tilde{\boldsymbol{a}}^c|c \sim Z_c \exp\frac{-\|\tilde{\boldsymbol{a}}-\tilde{\mu}_c\|_2^2}{2\tilde{\sigma}_c^2}, c \in \mathcal{Y}, \tag{A5}$$

$$\tilde{\boldsymbol{b}}^c|c \sim B_c \exp -\frac{\|\tilde{\boldsymbol{b}}^c - (\boldsymbol{\beta} - \gamma\frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}}-\tilde{\mu}_c}{\sqrt{\tilde{\sigma}_{\tilde{\boldsymbol{a}}}^2+\epsilon}})\|_2^2}{2\frac{\gamma^2\tilde{\sigma}_c^2}{\tilde{\sigma}_{\tilde{\boldsymbol{a}}}^2}} = \mathcal{N}(\tilde{\boldsymbol{a}}; \boldsymbol{\beta} - \gamma\frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}}-\tilde{\mu}_c}{\sqrt{\tilde{\sigma}_{\tilde{\boldsymbol{a}}}^2+\epsilon}}, \frac{\gamma\tilde{\sigma}_c}{\tilde{\sigma}_{\tilde{\boldsymbol{a}}}}), c \in \mathcal{Y}, \tag{A6}$$

where

$$\mu_c = \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x}|\text{argmax}\mathcal{F}(\boldsymbol{x})=c)}[\boldsymbol{a}], \tilde{\mu}_c = \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x}|\text{argmax}\tilde{\mathcal{F}}(\boldsymbol{x})=c)}[\tilde{\boldsymbol{a}}], \tag{A7}$$

$$\sigma_c = \text{Std}_{\boldsymbol{x}\sim p(\boldsymbol{x}|\text{argmax}\mathcal{F}(\boldsymbol{x})=c)}(\boldsymbol{a}), \tilde{\sigma}_c = \text{Std}_{\boldsymbol{x}\sim p(\boldsymbol{x}|\text{argmax}\tilde{\mathcal{F}}(\boldsymbol{x})=c)}(\tilde{\boldsymbol{a}}), c \in \mathcal{Y}, \tag{A8}$$

$$\mu_{\boldsymbol{a}} = \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\boldsymbol{a}], \tilde{\mu}_{\tilde{\boldsymbol{a}}} = \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\tilde{\boldsymbol{a}}], \sigma_{\boldsymbol{a}} = \text{Std}_{\boldsymbol{x}\sim p(\boldsymbol{x})}(\boldsymbol{a}), \tilde{\sigma}_{\tilde{\boldsymbol{a}}} = \mathbb{E}_{\boldsymbol{x}\sim p(\tilde{\boldsymbol{a}})}[\tilde{\boldsymbol{a}}]. \tag{A9}$$

For a sufficiently trained network, it is well-known that, with the neural collapse (Papyan et al., 2020), μ_c and $\sigma_c, c \in \mathcal{Y}$ form a simplex and are uniformly distributed. Specifically, in neural collapse scenarios, the features of each class form a simplex equiangular tight frame. This means that all features share (nearly) the same within-class variance and exhibit uniform mean values. Below, we try to find out the characteristics of a backdoored model.

A.1. Characterize the Backdoored Model

We denote the poisoned sample as $x(t) = x + G(x)$, where x is a benign input and $G(x)$, the trigger, is very small (*i.e.*, $\|G(x)\|_2 \ll 1$). The trigger $G(x)$ can be static or vary with different inputs, which fools the backdoored model $\tilde{\mathcal{F}}$ into recognizing the poisoned samples as the attacked target class t instead of its true class c. For clarity, we simplify the trigger $G(x)$ as $\boldsymbol{t}$. In this paper, we assume that all images have been normalized, *i.e.*, $x \in [0, 1]$. Accordingly, $||t||_2 \ll 1$ holds in practice since the triggers are either very sparse (*e.g.*, BadNets) or have a small overall magnitude (*e.g.*, WaNet). So the

feature distribution of $\boldsymbol{x}(\boldsymbol{t}) = \boldsymbol{x} + \boldsymbol{t}$ may be approximated by

$$\tilde{\boldsymbol{a}}(\boldsymbol{t}) \approx \tilde{\boldsymbol{a}}_l(\boldsymbol{x}+\boldsymbol{t}) = \tilde{\boldsymbol{a}}_l(\boldsymbol{x}) + \nabla \tilde{\boldsymbol{a}}_l(\boldsymbol{x})^T \boldsymbol{t} = \tilde{\boldsymbol{a}} + \nabla \tilde{\boldsymbol{a}}^T \boldsymbol{t}, \tag{A10}$$

$$\tilde{\boldsymbol{b}}(\boldsymbol{t}) = \gamma \left(\frac{\tilde{\boldsymbol{a}}(\boldsymbol{t}) - \tilde{\mu}_{\tilde{\boldsymbol{a}}}}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}} \right) + \boldsymbol{\beta} \approx \tilde{\boldsymbol{b}} + \frac{\nabla \tilde{\boldsymbol{a}}^T \boldsymbol{t}}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}} \equiv \tilde{\boldsymbol{b}} + \tilde{\boldsymbol{v}}^T \boldsymbol{t}. \tag{A11}$$

As $\boldsymbol{x}(\boldsymbol{t})$ should be recognized as category t, the conditional probability of $\tilde{\boldsymbol{b}}(\boldsymbol{t})$ being sampled from $\tilde{\boldsymbol{b}}|c$ should be smaller than from $\tilde{\boldsymbol{b}}|t$ for all $c \in \mathcal{Y}$. The assumption holds, particularly for the deeper hidden layers, under the Gaussian mixture distribution and a well-trained network. Specifically, in Equations (A3) and (A4), we assume the conditional distribution $a_l(x)|\arg\max f(x) = c$ to be Gaussian. This conditional distribution is derived after completing the forward pass and examining the previous layers, and it remains unchanged, *i.e.*, $p(a_l(x) \in a_l(A)|\arg\max f(x) = c) = p(a_{l+1}(x) \in a_{l+1}(A)|\arg\max f(x) = c), A = x : \arg\max f(x) = s$. Clearly, in the last layer, the probability of $a_l(x)$ belonging to class c will always be larger than other classes. Therefore, the assumption holds for the conditional distribution of $a_l(x)|\arg\max f(x) = c$. Thus, we can get, $\forall \boldsymbol{x}, \boldsymbol{t} \in \mathcal{X}$,

$$B_t \exp - \frac{\|\tilde{\boldsymbol{b}}(\boldsymbol{t}) - (\boldsymbol{\beta} - \gamma \frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}} - \tilde{\mu}_t}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}})\|_2^2}{2 \frac{\gamma^2 \tilde{\sigma}_t^2}{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}}}} > B_c \exp - \frac{\|\tilde{\boldsymbol{b}}(\boldsymbol{t}) - (\boldsymbol{\beta} - \gamma \frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}} - \tilde{\mu}_c}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}})\|_2^2}{2 \frac{\gamma^2 \tilde{\sigma}_c^2}{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}}}}, \forall c \in \mathcal{Y}, c \neq t, \tag{A12}$$

$$\Leftrightarrow \log \frac{B_t}{B_c} + \frac{1}{2\gamma^2 \tilde{\sigma}_t^2 \tilde{\sigma}_c^2} \left(\tilde{\sigma}_t^2 \|\tilde{\boldsymbol{b}}(\boldsymbol{t}) - (\boldsymbol{\beta} - \gamma \frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}} - \tilde{\mu}_c}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}} - \tilde{\sigma}_c^2)\|_2^2 - \tilde{\sigma}_c^2 \|\tilde{\boldsymbol{b}}(\boldsymbol{t}) - (\boldsymbol{\beta} - \gamma \frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}} - \tilde{\mu}_t}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}})\|_2^2 \right) > 0, \tag{A13}$$

$$\forall \tilde{\boldsymbol{b}}(\boldsymbol{t}) = \tilde{\boldsymbol{b}} + \tilde{\boldsymbol{v}}^T \boldsymbol{t}. \tag{A14}$$

Note that this is actually a *quadratic form* (the form of $ax^2 + bx + c > 0, \forall x$) of $\tilde{\boldsymbol{b}}$, to make sure the above inequality holds for all $\tilde{\boldsymbol{b}}$ (or at least most of $\tilde{\boldsymbol{b}}$ in the feature space), it is obvious that the quadratic coefficient $(\tilde{\sigma}_t^2 - \tilde{\sigma}_c^2)$ must be positive, so we should have

$$\tilde{\sigma}_t > \tilde{\sigma}_c, \forall c \in \mathcal{Y}, c \neq t. \tag{A15}$$

So we can confirm a key characteristic of the backdoored model, that the variance of the attacked target class t is larger than any of the others.

A.2. Parameter-oriented Scaling Consistency of Backdoored Models

After obtaining the above characteristic of the backdoored model, we can then prove the parameter-oriented scaling consistency of it.

Let

$$\Gamma_c = \boldsymbol{\beta} - \gamma \frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}} - \tilde{\mu}_c}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}}, c = 1, \cdots, C. \tag{A16}$$

Considering the above mixture of the Gaussian model, a sample $\boldsymbol{x}$ will be classified into class t if and only if

$$B_t \exp - \frac{\|\tilde{\boldsymbol{b}} - (\boldsymbol{\beta} - \gamma \frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}} - \tilde{\mu}_t}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}})\|_2^2}{2 \frac{\gamma^2 \tilde{\sigma}_t^2}{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}}}} > B_c \exp - \frac{\|\tilde{\boldsymbol{b}} - (\boldsymbol{\beta} - \gamma \frac{\tilde{\mu}_{\tilde{\boldsymbol{a}}} - \tilde{\mu}_c}{\sqrt{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}} + \epsilon}})\|_2^2}{2 \frac{\gamma^2 \tilde{\sigma}_c^2}{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}}}}, \forall c \in \mathcal{Y}, c \neq d, \tag{A17}$$

$$\Leftrightarrow \log \frac{B_t}{B_c} + \frac{\tilde{\sigma}^2_{\tilde{\boldsymbol{a}}}}{2\gamma^2 \tilde{\sigma}_t^2 \tilde{\sigma}_c^2} \left(\tilde{\sigma}_t^2 \|\tilde{\boldsymbol{b}} - \Gamma_c\|_2^2 - \tilde{\sigma}_c^2 \|\tilde{\boldsymbol{b}} - \Gamma_t\|_2^2 \right) \geq 0. \tag{A18}$$

The above can stand if

$$\log \frac{B_t}{B_c} + \frac{\tilde{\sigma}_{\tilde{\boldsymbol{a}}}^2}{2\gamma^2\tilde{\sigma}_t^2\tilde{\sigma}_c^2}\left((\tilde{\sigma}_t^2-\tilde{\sigma}_c^2)\|\tilde{\boldsymbol{b}}\|_2^2 - 2\|\tilde{\sigma}_t^2\Gamma_c-\tilde{\sigma}_c^2\Gamma_t\|_2\|\tilde{\boldsymbol{b}}\|_2 + \frac{\|\tilde{\sigma}_t^2\Gamma_c-\tilde{\sigma}_c^2\Gamma_t\|_2^2}{\tilde{\sigma}_t^2-\tilde{\sigma}_c^2}\right) \tag{A19}$$

$$+\frac{\tilde{\sigma}_{\tilde{\boldsymbol{a}}}^2}{2\gamma^2\tilde{\sigma}_t^2\tilde{\sigma}_c^2}\left(\tilde{\sigma}_t^2\|\Gamma_c\|_2^2-\tilde{\sigma}_c^2\|\Gamma_t\|_2^2-\|\tilde{\sigma}_t^2\Gamma_c-\tilde{\sigma}_c^2\Gamma_t\|_2^2\right)\geq 0 \tag{A20}$$

$$\Leftrightarrow \|\tilde{\boldsymbol{b}}\|_2 \geq \max\{\frac{1}{\sqrt{\tilde{\sigma}_t^2-\tilde{\sigma}_c^2}}\sqrt{-\tilde{\sigma}_t^2\|\Gamma_c\|_2^2+\tilde{\sigma}_c^2\|\Gamma_t\|_2^2+\|\tilde{\sigma}_t^2\Gamma_c-\tilde{\sigma}_c^2\Gamma_t\|_2^2-\log\frac{B_t}{B_c}\frac{2\gamma^2\tilde{\sigma}_t^2\tilde{\sigma}_c^2}{\tilde{\sigma}_{\tilde{\boldsymbol{a}}}^2}} \tag{A21}$$

$$+\frac{\|\tilde{\sigma}_t^2\Gamma_c-\tilde{\sigma}_c^2\Gamma_t\|_2}{\tilde{\sigma}_t^2-\tilde{\sigma}_c^2},0\}. \tag{A22}$$

So just like Equation (A12), the above is also a quadratic form for $\|\tilde{\boldsymbol{b}}\|_2$ with positive quadratic coefficient $(\tilde{\sigma}_t^2-\tilde{\sigma}_c^2)>0$. So when $\|\tilde{\boldsymbol{b}}\|_2$ is large enough (Equation (A22)), we will always have $\boldsymbol{x}$ is more likely to be identified into category t than all the others.

Remark A.1. Note that scale the parameter $\boldsymbol{\beta}_{\text{Inf}}, \gamma_{\text{Inf}}$ when inference will not influence the value of $\boldsymbol{\beta}, \gamma$ in Equation (A17). The $\boldsymbol{\beta}, \gamma$ in Equation (A17) is used to describe the underlying feature distributions (which are assumed to be the mixture of Gaussians). They will not change upon training finished.

As a result, when we scale the Batch Norm parameter $\boldsymbol{\beta}_{\text{Inf}}, \gamma_{\text{Inf}}$, we will get a $\tilde{\boldsymbol{b}}_{\text{Scale}}$ with larger norm propositional to γ_{Inf} and linearly increasing with respect to $\boldsymbol{\beta}_{\text{Inf}}$. When $\boldsymbol{\beta}_{\text{Inf}}, \gamma_{\text{Inf}}$ are larger enough, the scaled feature $\tilde{\boldsymbol{b}}_{\text{Scale}}$ will make Equation (A17) always positive.

Remark A.2. The above proof can be intuitively understood as follows: if we sample from a mixture of Gaussian distribution, then all remote points will be sampled from the Gaussian with the largest variance.

B. Detailed Configurations of the Empirical Study in Section 3

In this section, we adopt BadNets (Gu et al., 2017), WaNet (Nguyen & Tran, 2021), and BATT (Xu et al., 2023) as examples for our analysis. These attacks epitomize static, dynamic, and physical backdoor attacks, respectively. Our experiments are conducted on the CIFAR-10 dataset (Krizhevsky et al., 2009), using the ResNet18 model (He et al., 2016a). For each attack, we set the poisoning rate (ρ) to 0.1, achieving ASRs over 99%. In particular, we implement the backdoor attacks using their official codes with default settings. Specifically, the backdoor trigger for BadNets is represented as a 3×3 grid in black-and-white and is added to the lower-right corner of the poisoned images. For WaNet, the trigger is applied to the original images through elastic image warping transformation. In the case of BATT, the poisoned samples are obtained by rotating the original images by sixteen degrees. These attacks are implemented using the BackdoorBox toolkit (Li et al., 2023a)[1].

Regarding the scaling procedure, we adopt a layer-wise weight scaling operation to generate the parameter-amplified models. we scale up on the BN parameters (*i.e.*, γ and $\boldsymbol{\beta}$) with $\omega = 1.5$ times starting from the last layer and gradually moving forward to more layers. For example, in a 20-layer model, the first iteration involves scaling the weights of the 20th layer, and the next iteration extends the scaling to the 20th and the 19th layers, and so on. We then calculate the *average confidence* of 2000 testing samples for each parameter-scaled model. In this paper, *confidence* refers to the predicted probability assigned to an input sample for a specified label. For instance, if an image of a *cat* is predicted as the *cat* label with a probability of 0.9, then the *confidence* of the input under the *cat* label is 0.9. The average confidence is defined as the average probability of samples on the label predicted by the original unamplified model.

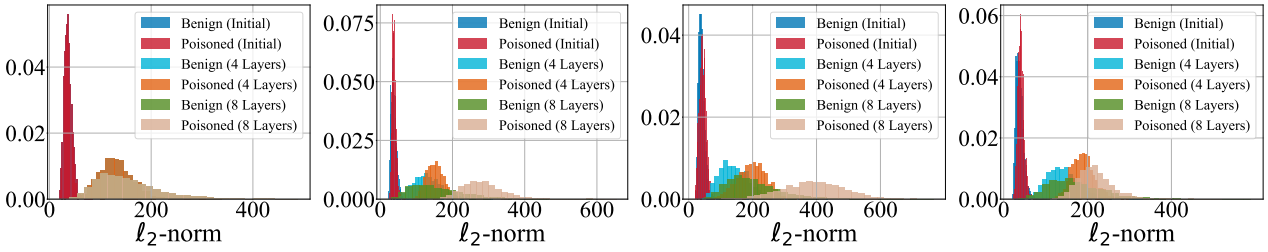
C. Detailed Exploration of amplifying a single BN layers in Section 3

As described in Section 3, we find amplifying only a single BN layer may require an unreasonably large amplification factor, and due to the nonlinearity of neural network layers, often leads to unstable defense performance across different attacks. To further explain the phenomenon, we conduct an empirical investigation aimed at investigating the percentage of benign samples to be predicted as the target class when amplifying the learnable parameters of individual BN layers with scale S. The results are displayed in Table Table A1, and we have three primary observations:

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

Table A1. The proportion (%) of benign samples in CIFAR-10 predicted to the target class when amplifying only a single BN layer.

Index →	1				5				15			
Scales S ↓	BadNets	WaNet	BATT	Ada-patch	BadNets	WaNet	BATT	Ada-patch	BadNets	WaNet	BATT	Ada-patch
5	96.75	10.50	62.86	0.00	92.43	93.25	5.04	12.85	11.37	99.32	99.13	76.81
10	100.00	53.53	38.81	0.00	100.00	100.00	2.19	27.40	16.33	100.00	100.00	89.66
100	100.00	100.00	100.00	0.15	100.00	100.00	99.96	91.56	27.40	100.00	100.00	96.10
1000	100.00	100.00	100.00	0.43	100.00	100.00	100.00	93.99	28.89	100.00	100.00	96.45
100000	100.00	100.00	100.00	0.44	100.00	100.00	100.00	94.18	29.01	100.00	100.00	96.49



(a) Benign Model (b) BadNets (c) WaNet (d) BATT

Figure A1. Histograms of the ℓ_2-norm of the final feature map of samples generated by models with different numbers of amplified BN layers. Increasing the number of amplified layers increases both value and variance of features.

Table A2. The performance (AUROC, F1) of our defense with forward model scaling process on the CIFAR-10 dataset. We mark the best result in boldface and failed cases (< 0.7) in red.

Metrics	BadNets	Blend	PhysicalBA	IAD	WaNet	ISSBA	BATT	SRA	LC	NARCISSUS	Adap-Patch
AUROC	0.997	0.678	0.964	0.999	0.910	0.998	0.635	0.952	0.450	0.941	0.960
F1	0.964	0.002	0.908	0.966	0.639	0.970	0.052	0.904	0	0.922	0.831

(1) The amplification factor for achieving effective defense varies considerably from layer to layer. (2) Some attacks (*e.g.*, WaNet and BATT) require an unreasonably large amplification factor to achieve a substantial misclassification rate. (3) Amplifying only a single BN layer may not be adequate to misclassify the majority of benign samples in some cases. For instance, amplifying the first BN layer alone cannot misclassify benign samples from the Ada-patch attack into the intended target class.

To address this, we spread the amplification across multiple consecutive BN layers, using a small factor (*e.g.*, 1.5) on each layer. Instead of controlling the layer-wise amplification factor, we vary the number of amplified layers to achieve different levels of accumulated amplification. This relation is demonstrated in Figure A1 (see Figure 3 for the density plot), where we see amplifying more layers induces higher last-layer activations, and increases the room to differentiate poisoned samples from the benign ones.

D. Why Scale the Later Layers?

Our defense relies on building a profile of how the target model behavior changes under progressive modifications to the model. Motivated by a widely accepted hypothesis (*e.g.*, (Tishby & Zaslavsky, 2015; Huang et al., 2022; Jebreel et al., 2023)) that layers situated towards the later stages exert a more direct influence on the ultimate model output, we designed our defense by amplifying the model parameters in stages, starting from the last hidden layer and progressively moving backward through the preceding layers. Here, we examine the alternative of a forward model scaling approach, which scales model parameters starting from the initial layers of the model and then progressing forward to the latter layers.

The results in Table A2 demonstrate that while this defense strategy proves to be effective against most backdoor attacks, such as BadNets, and ISSBA, it exhibits poor performance against others like Blend, BATT, and LC attacks. This discrepancy may be attributed to the fact that in those attacks, the trigger features closely resemble benign features in the model's shallow layers, making it challenging for the amplification operation to sufficiently separate these two types of features.

Table A3. The performance (AUROC, F1) of our defense with amplifying all of the BN layers on the CIFAR-10 dataset. We mark the best result in boldface and failed cases (< 0.7) in red.

Metrics	BadNets	Blend	PhysicalBA	IAD	WaNet	ISSBA	BATT	SRA	LC	NARCISSUS	Adap-Patch
AUROC	0.961	0.664	0.947	0.949	0.938	0.949	0.947	0.942	0.224	0.992	0.679
F1	0.949	0.060	0.926	0.952	0.941	0.951	0.940	0.943	0	0.938	0

E. Why not Amplifying All BN Layers?

In our defense, we amplify the later parts of the original model. It is motivated by the previous findings that trigger patterns often manifest as complicated features learned by the deeper (convolutional) layers of DNNs, especially for those attacks with elaborate designs (Huang et al., 2022; Jebreel et al., 2023). It is also consistent with our observations in Figure 2.

We investigate the performance of our defense by amplifying all BN layers within a model. As shown in Table A3, amplifying all layers leads to defense failure against Blend, LC, and WaNet attacks. In particular, its F1 score drops to 0, suggesting that amplifying all layers in the defense fails to detect any poisoned samples.

F. Detailed Settings for Experimental Datasets and Configurations

In line with the existing backdoor defense methods (Guo et al., 2023b; Liu et al., 2023; Gao et al., 2021), we select the most commonly used benchmark datasets and model architectures for our experiments. The datasets and models used are outlined in Table A4.

Table A4. The overview of the image datasets and the related classifiers used in our experiments.

Datasets	#Classes	Input Sizes	#Train. & Test. Images	Classifiers
CIFAR-10	10	$32 \times 32 \times 3$	50,000, 10,000	ResNet18, PreactResNet18, MobileNet
GTSRB	43	$32 \times 32 \times 3$	39,200, 12,600	ResNet18, PreactResNet18, MobileNet
SubImageNet-200	200	$224 \times 224 \times 3$	100,000, 10,000	ResNet18

CIFAR-10 is a benchmark dataset consisting of $3 \times 32 \times 32$ color images representing ten different object categories (Krizhevsky et al., 2009). The training set comprises 50,000 images, while the test set contains 10,000 images, with an equal distribution across the ten classes.

GTSRB is a benchmark dataset consisting of images of German traffic signs, categorized into 43 classes (Stallkamp et al., 2012). The training set consists of 39,209 images, while the test set contains 12,630 images. Given the considerable variation in image sizes within this dataset, we resize all images to a uniform size of $3 \times 32 \times 32$ for our experiments, ensuring consistency and convenience in handling.

SubImageNet-200. We adopt a subset of the ImageNet benchmark dataset (Deng et al., 2009) by randomly selecting 200 categories from the most common categories in the original ImageNet. Specifically, the subset includes 100,000 images from the original ImageNet for training (500 images per class) and 10,000 images for testing (50 images per class). For simplicity, all images are resized to a uniform dimension of $3 \times 224 \times 224$.

G. Details of Training Backdoored Models

G.1. Backdoor Attacks

In Section 5, we assess the effectiveness of our defense against thirteen backdoor attacks. These attacks are categorized into three types: **1)** poisoning-only attacks, **2)** training-controlled, **3)** and model-controlled attacks.

- **Poison-only Backdoor Attacks:** For the most commonly studied poisoning-only attacks, we consider various forms. This includes classic static attacks like **(1)** BadNet (Gu et al., 2017) and **(2)** Blend (Bai et al., 2021), sample-specific attack such as **(3)** ISSBA (Li et al., 2021a), clean-label attacks represented by **(4)** Label-Consistent (LC) (Turner et al., 2019) and **(5)** NARCISSUS (Zeng et al., 2023). In addition, we also consider adaptive attacks like **(6)** TaCT (Tang et al., 2021) and **(7)** Adap-Patch (Qi et al., 2023), which are designed to slip past existing defenses.

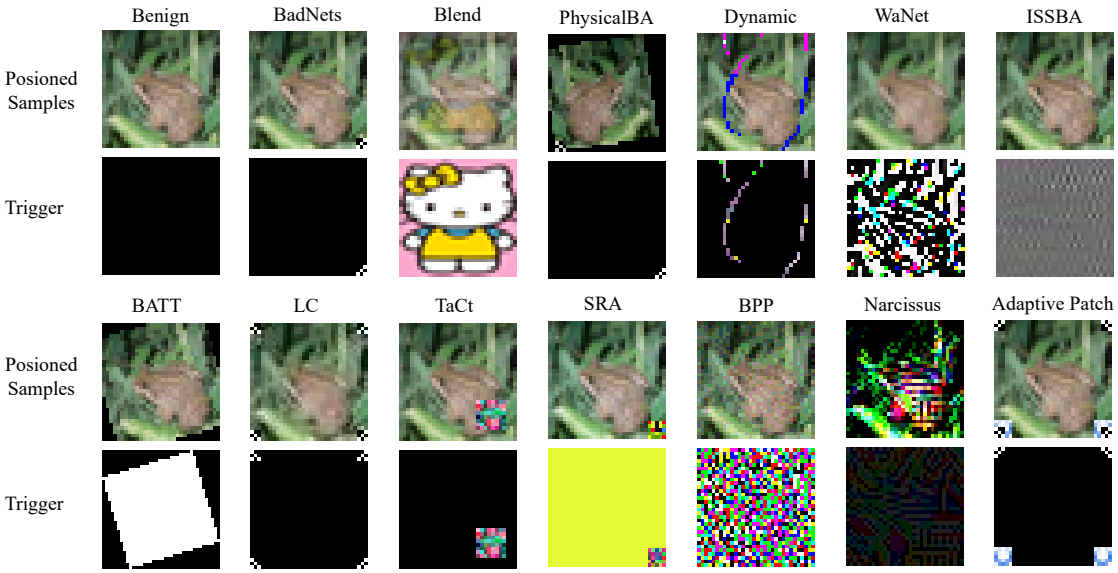


Figure A2. The examples of various triggers in the attacks adopted in our study and the corresponding poisoned samples.

- **Training-controlled Backdoor Attacks:** The training-controlled attacks include the **(8)** Dynamic (Nguyen & Tran, 2020), **(9)** WaNet (Nguyen & Tran, 2021), **(10)** BPP (Wang et al., 2022c) attacks, and physical backdoor attacks, including **(11)** PhysicalBA (Li et al., 2021c) and **(12)** BATT (Xu et al., 2023).

- **Model-controlled Backdoor Attacks:** we assess attacks involving direct modification of model parameters, such as **(13)** subnet replacement attack (SRA) (Qi et al., 2022).

The poisoning rate ρ for data-poisoning-based backdoor attacks is set to 0.1. The target class label is set to 0. In particular, the BATT attack consists of two attack modes, utilizing spatial rotation and translation transformations as triggers, respectively. In our study, we specifically employ spatial rotation as our triggers. The examples of both triggers and the corresponding poisoned samples are depicted in Figure A2.

G.2. Additional Details of Training Backdoored Models

We adopt the standard training pipeline for developing backdoor models. This involves an SGD optimizer with a momentum of 0.9 and a weight decay of 10^{-4}. The initial learning rate is set at 0.1, which is reduced to 10% of its previous value at the 50th and 75th epochs. The training comprises 200 epochs with a batch size of 128. For data augmentation on the CIFAR-10 dataset, we apply RandomHorizontalFlip and RandomCrop32 (randomly cropping images to a size of $3 \times 32 \times 32$). Additionally, RandomRotation15 is used to randomly rotate images within a range of [-15, 15] degrees.

For data augmentation on the CIFAR-10 dataset, we utilize RandomHorizontalFlip with a probability of 0.5 and RandomCrop32, which randomly crops images to a size of $3 \times 32 \times 32$. For the GTSRB dataset, we employ the RandomRotation15 augmentation technique, where images are randomly rotated within a range of [-15, 15] degrees. For the GTSRB dataset, we apply RandomCrop224, RandomHorizontalFlip, and RandomRotation20 to enhance the accuracy of the backdoored model on the benign samples.

All experiments are performed on a server with the Ubuntu 16.04.6 LTS operating system, a 3.20GHz CPU, 2 NVIDIA's GeForce GTX3090 GPUs with 62G RAM, and an 8TB hard disk.

G.3. Effectiveness of the Backdoored Attacks

Following the settings in existing backdoor attacks, we use two metrics to measure the effectiveness of the backdoor attacks: attack success rate (ASR) and benign accuracy (BA). ASR indicates the success rate of classifying the poisoned samples into the corresponding target classes. BA measures the accuracy of a backdoored model on the benign testing dataset.

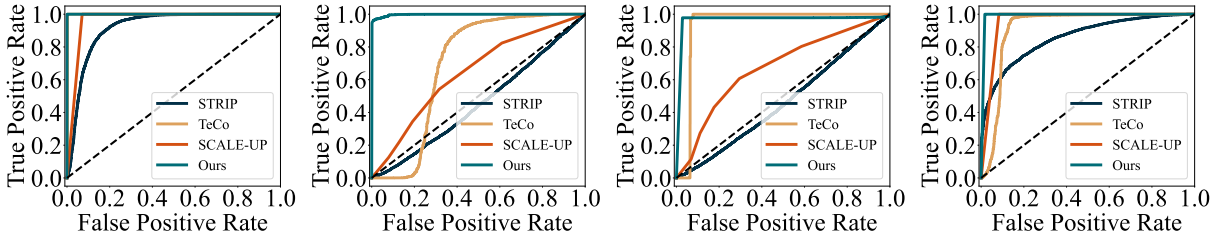
BA and ASR for different backdoor attacks are included in Table A5 and Table A6.

Table A5. The performance (BA, ASR) on different attacks and datasets with ResNet18 model.

Datasets	BadNets		Blend		PhysicalBA		Dynamic		WaNet		ISSBA		BATT	
	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR
CIFAR10	0.929	1	0.931	0.999	0.937	0.966	0.938	1	0.948	0.997	0.936	1	0.939	1
GTSRB	0.976	0.998	0.966	1	0.976	0.968	0.971	1	0.994	0.997	0.968	1	0.979	0.998
SubImageNet-200	0.808	0.998	0.823	0.998	0.796	0.994	0.793	1	0.768	0.967	0.803	0.990	0.695	0.997

Table A6. The performance (BA, ASR) on other backdoor attacks with the ResNet18 model on the CIFAR-10 dataset.

Metrics	LC	TaCT	SRA	BPP	NARCISSUS	Adap-Patch
BA	92.28	93.78	88.98	89.68	89.80	93.54
ASR	100	99.00	99.90	99.70	96.92	99.89



(a) BadNets (b) Blend (c) WaNet (d) BATT

Figure A3. ROC curves of defenses against different attacks on the CIFAR-10 dataset.

H. Implementation of the Baseline Defenses

(1) STRIP: We implement STRIP following their official open-sourced codes[2]. STRIP detects backdoor attacks by observing the prediction behaviors of an input sample when superimposing benign features on it.

(2) TeCo: We implement TeCo following their official open-sourced codes[3].

(3) SCALE-UP: We implement SCALE-UP (data-limited) following the most commonly used open-sourced toolbox codes[4].

I. Generalizability to Other Model Architectures

We evaluate the effectiveness of our defense on additional model architectures including PreActResNet18 (He et al., 2016b), and MobileNet (Krizhevsky et al., 2009). The defense performance is presented in Table A7. As shown, most of the average AUROC and F1 scores on both architectures are above 0.96, with a few slightly lower scores (still above 0.93). This result indicates that our defense has general applicability across different model architectures.

J. ROC Curve Comparison with Baseline Defenses

In addition to AUROC and F1 scire metrics, we also visually compare the ROC curves of competing defense methods against attacks. ROC curves for the CIFAR-10 experiments can be found in Figure A3.

K. Performance of Our IBD-PSC Against Additional Backdoor Scenarios

Table A8 presents the performance (AUROC and F1 scores) of our IBD-PSC against some other types of backdoor attacks, including clean-label attacks (LC (Turner et al., 2019), NARCISSUS (Zeng et al., 2023)), source-specific attack (TaCT (Tang et al., 2021)), training-controlled attack (BPP (Wang et al., 2022c)), model-controlled attack (SRA (Qi et al., 2022)), and adaptive attack (Adap-Patch (Qi et al., 2023)). The results demonstrate that IBD-PSC consistently outperforms other defense strategies across almost all types of backdoor attacks. It achieves the highest average scores in both AUROC and F1 metrics, marked in bold, underscoring its superior detection capabilities. This comprehensive evaluation affirms the robustness of

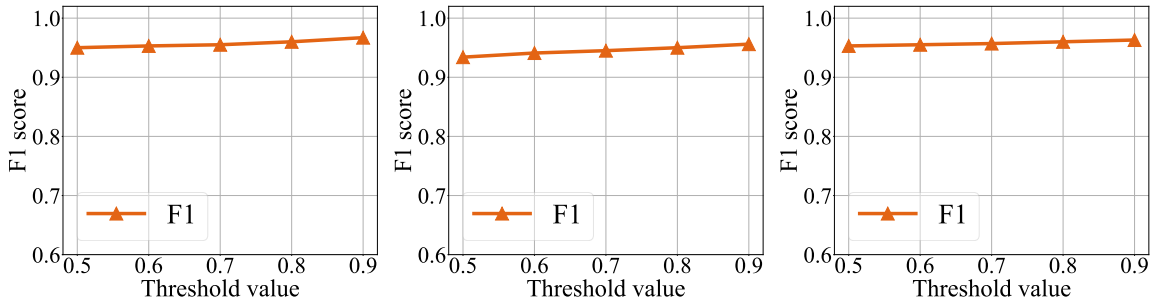
[2] https://github.com/garrisongys/STRIP
[3] https://github.com/CGCL-codes/TeCo
[4] https://github.com/vtu81/backdoor-toolbox

Table A7. The performance (AUROC, F1) of our defense on other model architectures.

Datasets	Models→	PreactResNet18		MobileNet		Avg.	
	Attacks↓	AUROC	F1	AUROC	F1	AUROC	F1
CIFAR10	BadNets	0.978	0.931	0.970	0.943	0.974	0.937
	IAD	0.989	0.965	0.969	0.951	0.979	0.958
	WaNet	0.977	0.949	0.937	0.940	0.957	0.945
	BATT	0.972	0.958	0.951	0.953	0.962	0.956
Datasets	Models→	PreactResNet18		MobileNet		Avg.	
	Attacks↓	AUROC	F1	AUROC	F1	AUROC	F1
GTSRB	BadNets	0.970	0.971	0.969	0.971	0.970	0.971
	IAD	0.970	0.970	0.966	0.966	0.968	0.968
	WaNet	0.964	0.933	0.986	0.977	0.975	0.955
	BATT	0.968	0.970	0.970	0.957	0.969	0.964

Table A8. Performance (AUROC, F1) of our IBD-PSC against various backdoor attacks including clean-label, source-specific, training-controlled, model-controlled, and adaptive attacks. We mark the best result in boldface and failed cases (< 0.7) in red.

Attacks→	LC		TaCT		SRA		BPP		NARCISSUS		Adap-Patch		Avg.	
Defenses↓	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1
STRIP	0.668	0.541	0.431	0.106	0.550	0.213	0.331	0.081	**0.952**	**0.949**	0.858	0.715	0.632	0.434
TeCo	0.818	0.685	**1.000**	0.946	0.933	0.919	**0.992**	0.926	0.927	0.864	0.947	0.948	0.940	0.908
SCALE-UP	0.943	**0.912**	0.614	0.234	0.580	0.453	0.860	0.832	0.673	0.000	0.941	0.913	0.754	0.496
IBD-PSC	**0.980**	0.834	0.986	**0.974**	**0.976**	**0.943**	0.990	**0.968**	0.939	0.924	**0.999**	**0.961**	**0.978**	**0.944**



(a) BadNets (b) WaNet (c) BATT

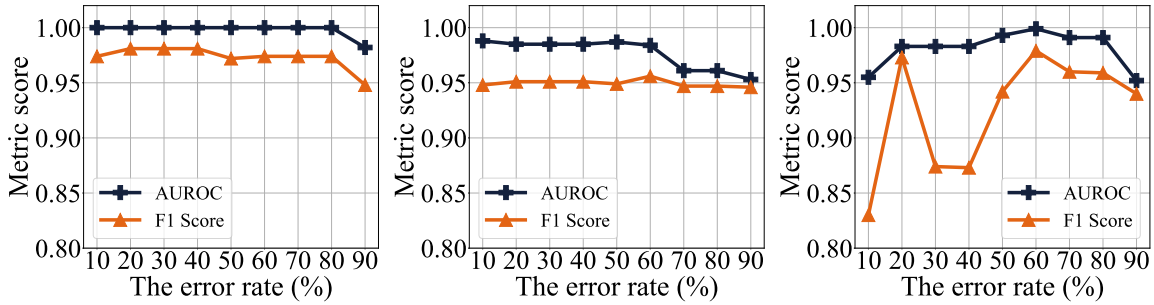
Figure A4. Impact of the value of threshold T on defense effectiveness.

IBD-PSC as a formidable defense mechanism in the ever-evolving landscape of backdoor attacks in cybersecurity.

L. Settings for the Inference Time Comparison

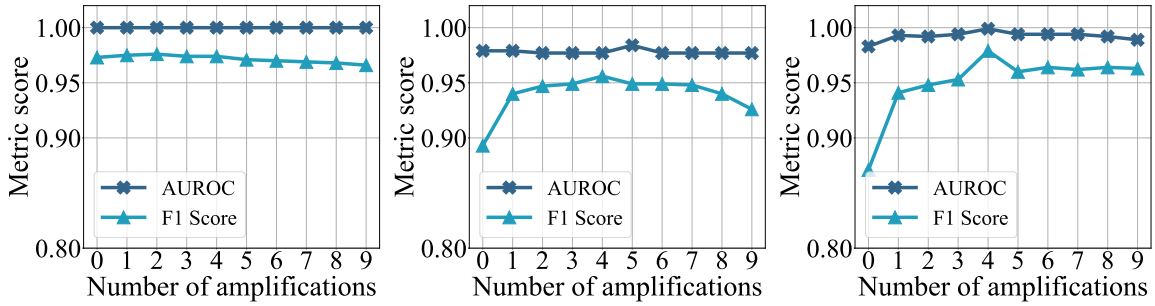
The inference time is critical for this task (*i.e.*, detecting poisoned testing images) because the detection is usually deployed as the 'firewall' for online inference. In the case of STRIP, TeCo, SCALE-UP, and our defense, defenders utilize the target model's prediction for defense purposes. This means that both detection and prediction can be carried out simultaneously.

We calculate the inference time of all defense methods under identical and ideal conditions to evaluate efficiency. For example, we assume that defenders will load all required models and images simultaneously, which demands more memory requirements compared to the standard model inference. This comparison is fair and reasonable due to the significant differences in mechanisms and requirements among the various defenses. More precisely, before inference, we engage in preparatory steps such as selecting the BN layers to be amplified and preparing the parameter-amplified models. These models are subsequently deployed across different machines, enabling simultaneous processing of input samples. While this approach requires additional storage space to accommodate the various model versions, it considerably accelerates the detection process. For SCALE-UP, we calculate the inference time needed to obtain predictions for multiple augmented images associated with a given input. This is achieved by concurrently feeding all the images into the deployed model as a batch instead of predicting them individually.



(a) BadNets (b) WaNet (c) BATT

Figure A5. The impact of error rate ξ on defense effectiveness.



(a) BadNets (b) WaNet (c) BATT

Figure A6. Impact of the number of amplifications (n) on defense effectiveness.

M. Ablation Studies

M.1. Impact of the Threshold T

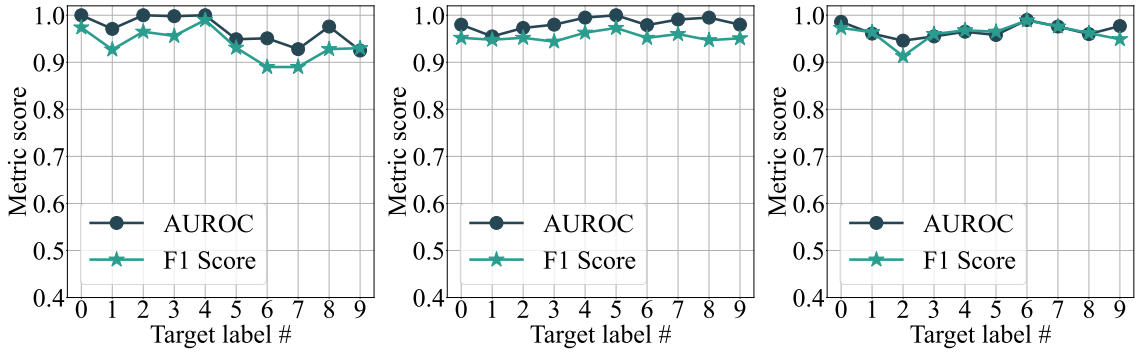
In our defense, we assess whether an input sample is malicious by comparing its PSC value to a predefined threshold T. Following the other experiments, we conduct an ablation study of T on three representative attacks: BadNets, WaNet, and BATT on the CIFAR-10 dataset, by adjusting T from 0.5 to 0.9. The results are shown in Figure A4. As we can see, a wide range of values of T can lead to a high F1 score. In our experiment, we set T to 0.9.

M.2. Impact of the Hyperparameter ξ

In our defense, we design an adaptive algorithm to dynamically select a suitable number of the BN layers to be amplified. The algorithm uses a predefined hyperparameter error rate threshold ξ. Here we empirically show that our defense is insensitive to changes in ξ. Again, this is demonstrated on three representative attacks: BadNets, WaNet, and BATT, with varying values of ξ from 10% to 90%. Figure A6 shows that the defense performance against BadNets and WaNet attacks exhibits remarkable resilience to variations in ξ. While the BATT attack does manifest a more pronounced response to changes in ξ, with the F1 score experiencing fluctuations, the metric is eventually higher when the error rate reaches approximately 60%. This observation signals that the overall influence of the error rate on the defense efficacy remains limited. Consequently, we advocate for an error rate of around 60%, as it appears to strike a judicious balance, ensuring adequate detection accuracy without unduly compromising the defense strategy against the assessed backdoor threats.

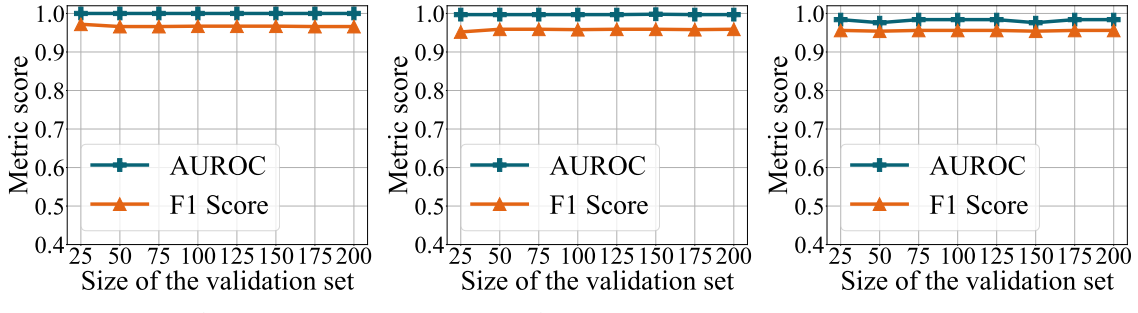
M.3. Impact of the Number of Amplified Models n

In our defense, we build a profile of n progressively amplified models, to capture the model's dynamic response to such interventions. In practice, the number of amplifications n is a defender-assigned hyper-parameter. As illustrated in Figure A6, the detection performance under the BadNets attack exhibits consistency across various values of n, suggesting a relative insensitivity to the number of amplifications. In contrast, for the WaNet and BATT attacks, there is an improvement in detection effectiveness as n increases, which plateaus when n reaches five. This stabilization suggests an optimal defense performance, and thus we establish $n = 5$ as the optimal value for our defense, ensuring stable detection performance.

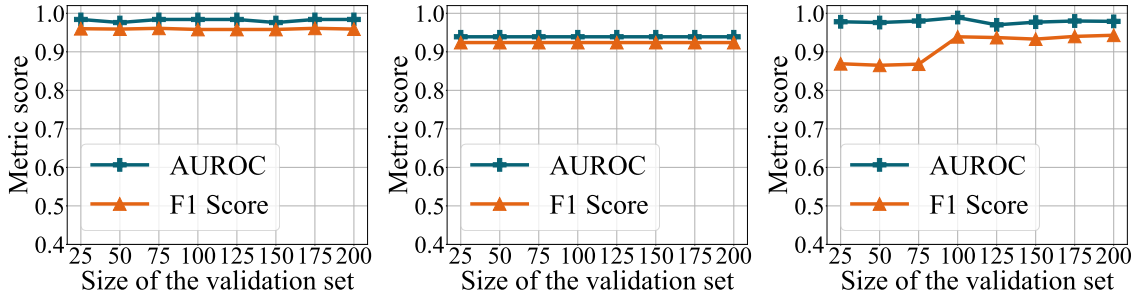


(a) BadNets (b) IAD (c) WaNet

Figure A7. Performance of our defense across 10 target labels of CIFAR10.



(a) BadNets (b) IAD (c) WaNet



(d) BATT (e) NARCISSUS (f) SRA

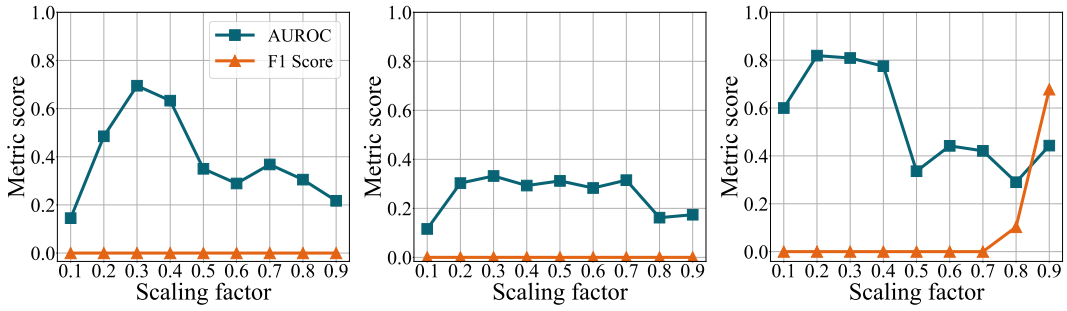
Figure A8. The impact for the size of the validation set.

Table A9. The false positive rate (%) of our defense on benign models on the CIFAR-10 dataset.

Dataset	Model 1	Model 2	Model 3	Model 4	Model 5
CIFAR-10	2.990	3.00	2.540	2.240	2.100
GTSRB	6.050	6.270	6.950	6.310	6.150
SubImageNet-200	0.290	0.370	0.370	0.320	0.320

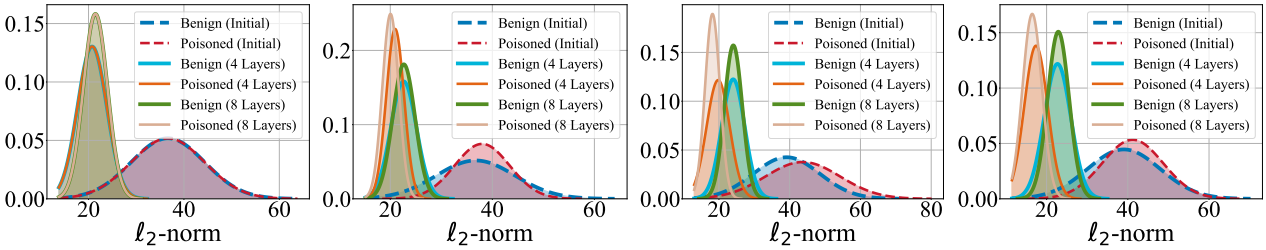
M.4. Impact of the Target Class

We further evaluate the robustness of our defense to the changes of the target class. We select three attacks, including the patch-based, dynamic, and physical backdoor attacks mentioned above, and apply them to target each of the ten labels of CIFAR-10. We display the AUROC and F1 scores of our defense against these backdoored models in Figure A7. As shown, our defense demonstrates consistent performance against different attacks and target labels. Specifically, the AUROC and F1 scores are consistently close to 1, with the average AUROC and F1 scores of each attack all exceeding 0.96 and 0.94, respectively. This indicates that our defense maintains strong performance against different types of attacks and target labels. Additionally, the standard deviations of AUROC and F1 scores across different cases are generally below 0.02.



(a) BadNets (b) WaNet (c) BATT

Figure A9. The impact of scaling factor (smaller than 1.0) on defense effectiveness.



(a) Benign Model (b) BadNets (c) WaNet (d) BATT

Figure A10. Distribution of L2 Norm Values for the last-hidden-layer activations under reduced model parameters (*i.e.*, magnification factors less than 1), where "n Layers" represents the number of scaled layers, counting backward from the model's final layer.

M.5. Impact of the Size of Local Benign Samples $\mathcal{D}_r$

Following similar studies in the literature (Wang et al., 2019; Guo et al., 2023b), we assume that defenders possess a small benign dataset $\mathcal{D}_r$ to calibrate different model parameters. By default, $\mathcal{D}_r$ contains merely 100 samples. We evaluate the robustness of our defense to the change in the size of $\mathcal{D}_r$. The results on the CIFAR-10 dataset using ResNet18 against six attacks on the CIFAR-10 dataset are shown in Figure A8. It is evident that for most attacks, including BadNets, IAD, WaNet, BATT, and NARCISSUS, the detection performance remains consistently high and relatively stable across varying sizes of $\mathcal{D}_r$. For the SRA attack, the F1 score increases and stabilizes when the size of $\mathcal{D}_r$ reaches 100. Overall, this demonstrates that our defense is effective with as few as 100 benign samples.

M.6. False Positive Rates on Benign Models

In this study, we investigate a scenario in which a defender obtains a third-party DNN model but cannot determine whether the model is compromised with backdoors. To ensure security, it is common to deploy an input-level backdoor detection system, similar to a network firewall, to filter potentially poisoned samples. In such a context, evaluating the impact of the deployed defenses on benign models is crucial.

To achieve this objective, we train five benign models using different random seeds. Subsequently, we conduct tests on these models to calculate the false positive rate of our defense, which represents the proportion of benign samples incorrectly identified as backdoor samples. These benign samples are incorrectly rejected by the defense during inference. Table A9 presents the false positive rates of our defense on different benign models trained on various datasets. We can observe variations in the false positive rate among different models, but overall, it remains relatively low (below 3% on the CIFAR-10 and 1% on the SubImageNet-200 dataset, and around 6% on the GTSRB dataset).

We attribute the higher false positive rate on GTSRB to the relative simplicity of image features in this dataset, making the models more prone to overfitting. Consequently, when amplifying model weights, some benign samples may not decrease their prediction confidence due to overfitting. This overfitting phenomenon on GTSRB has also been reported in the SCALE-UP defense (Guo et al., 2023b). However, in the real world, datasets are often more similar to the ImageNet dataset, characterized by its comprehensive and rich feature information. Our defense performs best on SubImageNet-200, achieving an error rate of less than 1%.

Table A10. The performance of our defense in defending against backdoor attacks with different poisoning rates on CIFAR-10.

Attacks →	BadNets			WaNet			BATT		
$\rho \downarrow$	ASR	AUROC	F1	ASR	AUROC	F1	ASR	AUROC	F1
0.5%	1.000	0.955	0.950	0.039	0.936	0.927	0.961	0.913	0.927
1%	1.000	0.950	0.951	0.286	0.953	0.935	0.993	0.948	0.948
2%	1.000	0.999	0.928	0.968	0.966	0.944	0.999	0.999	0.972
4%	1.000	0.998	0.912	0.972	0.977	0.959	1.000	1.000	0.967
6%	1.000	0.999	0.961	0.994	0.978	0.955	1.000	0.998	0.942
8%	1.000	1.000	0.981	0.996	0.983	0.960	1.000	0.985	0.958
10%	1.000	1.000	0.967	0.997	0.984	0.956	1.000	0.999	0.979

Table A11. The performance of our defense in defending against attacks with different poisoning rates (ρ) on SubImageNet-200.

Attacks →	BadNets			WaNet			BATT		
$\rho \downarrow$	ASR	AUROC	F1	ASR	AUROC	F1	ASR	AUROC	F1
2%	0.955	0.999	0.905	0.004	-	-	0.981	0.999	0.998
4%	0.960	1.000	0.996	0.182	0.944	-	0.967	1.000	0.999
6%	0.972	0.991	0.989	0.786	1.000	0.996	0.981	0.999	0.997
8%	0.974	0.997	0.995	0.818	0.986	0.976	0.980	1.000	0.999
10%	0.998	1.000	0.992	0.967	0.967	0.981	0.997	0.998	0.998

M.7. Is Shrinking as Effective as Amplifying?

In this study, we focus on detecting poisoned samples by amplifying model parameters with a scaling factor greater than one. To complement this approach, we conducted ablation experiments in this section involving shrinking model parameters using a scaling factor smaller than one.

According to Theorem 3.1, larger enough feature norms can induce a decrease in confidence for the original predicted class, if the inputs are benign samples (and certain classical assumptions in learning theory are adopted). Poisoned samples, instead, will stay fine. Therefore, by inversely reducing the values of parameters, we expect to observe a degradation in detection performance. The experiments are conducted across a range of reduced magnification factors (0.1 to 0.9) against BadNet, WaNet, and BATT attacks on the CIFAR-10 dataset using the ResNet18 model. The results displayed in Figure A9, clearly indicate a reduction in detection performance, as evidenced by lowered AUROC values and F1 scores approaching zero. This trend remains consistent across the various attack methods examined.

The reduction in detection performance with decreased parameter values reveals the effectiveness of parameter amplification as a defensive strategy, offering a reason to adopt this approach in safeguarding against backdoor threats.

We also examine the L2 Norm for the last-hidden-layer activations under reduced model parameters. We set the parameter reduction factor to 0.9 and reduced the values of parameters of the model's last four and eight hidden layers, respectively. The L2 norm is calculated on both benign models and backdoored models under BadNet, WaNet, and BATT attacks. As shown in Figure A10, a greater reduction in parameters led to a smaller L2 norm in both benign and backdoored models.

This observation provides empirical justification for the necessity of parameter amplification, thereby reinforcing our insights for our proposed defense.

N. Robustness Against Adaptive Attacks

N.1. Existing Attacks with Small Poisoning Rates

We evaluate the robustness of our defense against backdoor attacks with low poisoning rates. Specifically, we focus on three representative attacks (BadNets, WaNet, and BATT) on the CIFAR-10 dataset with poisoning rates (ρ) ranging from 0.5% to 10%, and on the SubImageNet-200 dataset with ρ from 2% to 10%, ensuring that most ASRs exceed 80%. The results presented in Table A10 and Table A11 demonstrate that our defense remains effective under low poisoning rates, achieving AUROC and F1 scores well above 0.9, even at a poisoning rate as low as 0.5%. We also compare the performance of our defense with baseline defenses under these conditions. As shown in Table A12, our defense outperforms the baseline defenses in most scenarios across different poisoning rates.

Table A12. Comparision of the performance in defending against backdoor attacks with different poisoning rates (ρ) on CIFAR-10.

$\rho \rightarrow$		2%		4%		6%		8%		10%	
Defenses ↓	Attacks ↓	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1
STRIP	BadNets	0.881	0.679	0.895	0.752	0.868	0.657	0.769	0.429	0.931	0.824
	WaNets	0.493	0.137	0.485	0.138	0.479	0.131	0.466	0.116	0.469	0.125
	BATT	0.779	0.579	0.650	0.364	0.656	0.385	0.808	0.639	0.449	0.258
TeCo	BadNets	1.000	0.916	0.997	0.952	0.981	0.929	0.994	0.937	0.998	0.970
	WaNets	0.992	0.891	0.976	0.905	0.999	0.944	0.906	0.945	0.923	0.915
	BATT	0.803	0.683	0.814	0.684	0.809	0.685	0.871	0.685	0.914	0.673
SCALE-UP	BadNets	0.959	0.918	0.964	0.914	0.959	0.910	0.971	0.915	0.962	0.913
	WaNets	0.746	0.698	0.766	0.726	0.730	0.624	0.689	0.646	0.672	0.529
	BATT	0.944	0.880	0.968	0.868	0.957	0.907	0.968	0.871	0.959	0.911
Ours	BadNets	0.999	0.928	0.998	0.912	0.999	0.961	1.000	0.981	1.000	0.967
	WaNets	0.966	0.944	0.977	0.959	0.978	0.955	0.983	0.960	0.984	0.956
	BATT	0.999	0.972	1.000	0.967	0.998	0.942	0.985	0.958	0.999	0.979

Table A13. The comparison of our defense and the baseline defenses in defending against the WaNet attack under poisoning rates of 0.5% and 1% on CIFAR-10.

ρ (%) →	0.5			1		
Defenses ↓	AUROC	TPR	FPR	AUROC	TPR	FPR
STRIP	0.403	0.039	0.100	0.421	0.033	0.100
TeCo	1.000	1.000	0.156	1.000	1.000	0.068
SCALE-UP	0.461	0.440	0.389	0.467	0.424	0.345
Ours	0.936	0.791	0.864	0.953	1.000	0.129

We observe that the ASRs (excluding testing samples from the target class) for WaNet are 3.9% and 28.6% at poisoning rates of 0.5% and 1%, respectively. This indicates that the attack nearly fails at a 0.5% poisoning rate. Following the suggestion from the Backdoor-Toolbox, we remove samples containing trigger patterns that still cannot be correctly predicted as the target label by backdoored DNNs. As shown in Table A13, our defense remains highly effective in these cases, although its performance is slightly lower than that of TeCo, which requires significantly more inference time. In contrast, both STRIP and SCALE-UP fail.

N.2. Adaptive Attacks in the Worst-case Scenario

In addition to the existing attacks, we further consider the worst-case scenario where potential adaptive attacks are tailored for our defense.

Design 1. A natural assumption is that the adversary would design an adaptive loss $\mathcal{L}_{\text{ada}} = \sum_{i=1}^{|\mathcal{D}_b|} \mathcal{L}(\hat{\mathcal{F}}_k^{\omega}(\boldsymbol{x}_i; \hat{\boldsymbol{\theta}}), y_i)$ to ensure that the benign samples are correctly predicted when subjected to model parameter amplification, hence breaking our consistency assumption. This adaptive loss is then integrated into the overall loss function as $\mathcal{L} = \alpha\mathcal{L}_{\text{bd}} + (1-\alpha)\mathcal{L}_{\text{ada}}$, where α represents the weighting factor.

The adversary would aim to find a α value that best balances the ASRs and the BAs. Table A14 presents the performance (BA, ASR) of the adaptive attacks under various α settings. As evident from the results, all three attacks (BadNets, WaNet, BATT) on the CIFAR-10 dataset employed in the experiments consistently exhibit high ASRs and BA across different values of α on the CIFAR-10 dataset, underscoring the effectiveness of the adaptive attacks.

On the other hand, we have shown in Table 5 such as adaptative attacks can still be effectively defended by our method. We conducted further investigation. We observed that the adaptive loss indeed induced a model $\mathcal{F}'$ substantially different from the nonadaptive version $\mathcal{F}$. However, our defense, particular Algorithm 1, can readjust for the modified backdoor model (Note that model $\mathcal{F}$ is an input in Algorithm 1). In particular, we observed that on a nonadaptive backdoor model $\mathcal{F}$, Algorithm 1 returns $k = 10$. On the adaptive model $\mathcal{F}'$, Algorithm 1 returns $k = 15$. In other words, our algorithm learned to exploit the earlier layers not touched by the adaptive attack. This ability to counter adaptive attacks is a key advantage of our method compared with the input-based SCALEUP method.

Design 2. We can also design another form of adaptive attack to mitigate the impact of parameter amplification. Specifically, we aim to reduce the confidence with which parameter-amplified models predict poisoned samples as belonging to the target

Table A14. The attack performance (BA, ASR) with the adaptive attack settings in "Design 1".

Weight→	0.2		0.5		0.9		0.99	
Attacks↓	BA	ASR	BA	ASR	BA	ASR	BA	ASR
BadNets	0.775	0.992	0.858	0.985	0.881	0.995	0.891	0.996
WaNet	0.906	0.948	0.891	0.977	0.877	0.935	0.879	0.813
BATT	0.851	0.986	0.846	0.994	0.840	0.983	0.831	0.981

Table A15. The attack performance (BA, ASR) of the adaptive attack in "Design 2" and the detection performance (AUROC, F1) of IBD-PSC against the adaptive attack on CIFAR-10. We mark the failed cases (where $BA < 70\%$) in red, given that the accuracy of models unaffected by backdoor attacks on clean samples is 94.40%.

$\alpha' \rightarrow$	0.01		0.1		0.5	
Attacks↓	BA / ASR	AUROC /F1	BA / ASR	AUROC / F1	BA / ASR	AUROC/ F1
BadNets	0.832 / 0.887	0.877 / 0.924	0.802 / 0.874	0.874 / 0.861	0.101/ 0.997	- / -
WaNet	90.88 / 99.87	0.999 / 0.956	87.07 / 99.15	0.985 / 0.934	85.16 / 89.10	0.887 / 0.895
BATT	0.745 / 0.997	0.996 / 0.982	0.648 / 0.998	- / -	0.463 / 0/994	- / -

Table A16. The proportion (%) of misclassified benign samples classified by the model on each category. In our cases, the target label is 0.

Attacks ↓, Labels →	0	1	2	3	4	5	6	7	8	9
BadNets (Original)	14.68	5.71	11.09	22.02	10.77	14.19	6.36	3.92	5.55	5.71
BadNets (Adaptive)	99.57	0.42	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
BATT (Original)	10.81	5.50	13.18	22.27	10.62	13.18	6.23	5.88	5.78	6.54
BATT (Adaptive)	94.67	0.02	0.30	3.84	0.91	0.09	0.06	0.00	0.02	0.09

class. Inspired by label smoothing, we design an adaptive loss term $\mathcal{L}'_{\text{ada}}$ to decrease the confidence of poisoned samples when model parameter amplification occurs. The adaptive loss term $\mathcal{L}'_{\text{ada}}$ is defined as:

$$\mathcal{L}'_{\text{ada}} = \sum_{j=1}^{|\mathcal{D}_p|} \mathcal{L}(\hat{\mathcal{F}}_k^{\omega}(\boldsymbol{x}_j; \hat{\boldsymbol{\theta}}), \hat{y}_i), \tag{A23}$$

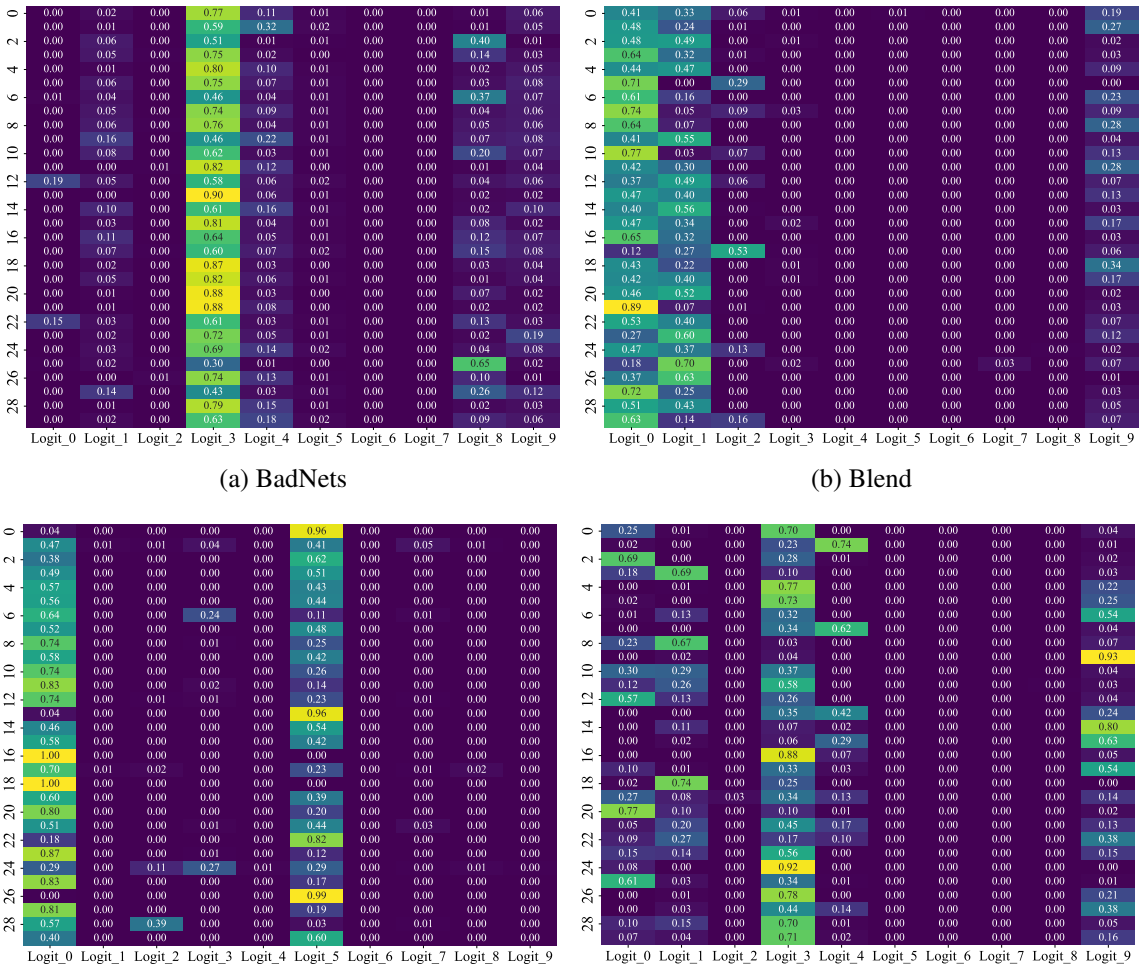
where $\hat{y}_i$ represents the label-smoothing form of t, and $\mathcal{D}_p$ denotes the set of poisoned samples. $\hat{y}_i$ is defined as:

$$\hat{y}_{i,c} = \begin{cases} 1 - \zeta & \text{if } c = t \\ \frac{\zeta}{C-1} & \text{otherwise.} \end{cases} \tag{A24}$$

Here, ζ is set to 0.2, specifically chosen to reduce the confidence with which poisoned samples are classified into the target class. The term C denotes the total number of classes, $|\mathcal{D}_p|$ represents the number of poisoned samples in the training set, and $\boldsymbol{x}_j$ denotes a poisoned sample.

We integrate the adaptive loss term $\mathcal{L}'_{\text{ada}}$ with the vanilla backdoor loss $\mathcal{L}_{\text{bd}}$ to formulate the overall loss function as $\mathcal{L}' = \alpha\mathcal{L}_{\text{bd}} + (1 - \alpha')\mathcal{L}'_{ada}$, where α' is a weighting factor. We evaluate the robustness of our defense under the same settings as described in Section 5.4. As shown in Table A15, reducing the confidence of poisoned samples significantly decreases BA, making the attack more noticeable. Moreover, our defense remains effective even against this new adaptive attack. The effectiveness of IBD-SPC largely stems from our adaptive layer selection strategy, which dynamically identifies BN layers for amplification. This approach ensures the robustness of our defense mechanism across various scenarios, whether the model is vanilla or adaptively backdoored.

We claim that the reduction of BA caused by the adaptive attack is mainly because the DNNs link both benign features and trigger features to the target class when using label smoothing, although the connection between trigger features and the target class is stronger. Specifically, attacked models tend to overfit trigger features when high confidence (*e.g.*, 1) is applied on poisoned samples, as seen in vanilla backdoor attacks. However, after label smoothing, the attacked models rely on both trigger and other features, as the task becomes more complicated and harder to fit accurately. Consequently, the adaptive-attacked model is more likely to predict benign samples as the target class, resulting in relatively low benign accuracy. To further verify this, we calculate the distribution of misclassified benign samples. As shown in Table A16, almost all misclassified benign samples are predicted as the target label (*i.e.*, 0) instead of other classes.



(a) BadNets (b) Blend

(c) BATT (d) Ada-patch

Figure A11. The heatmap of the average prediction confidences of benign samples within the target class predicted by the n scaled models on the CIFAR-10 dataset. The threshold of our defense is set at 0.9; samples with PSC values above this threshold are classified as poisoned. In our cases, the target label is 0.

We also design an adaptive attack variant that directly classifies poisoned samples to their ground-truth labels under parameter amplification. However, this approach imposes excessive regularization, preventing the learning of backdoors. This phenomenon aligns with our findings in Theorem 3.1. Additionally, we design another adaptive loss term by making parameter-amplified models categorize poisoned images as a different class (rather than the target class). However, even with a very small trade-off hyper-parameter for the adaptive loss term, it significantly decreases the BA of the attacked model by more than 30%.

O. The Impact of Parameter Amplification on the Benign Samples from Target class

To better understand the reduction in confidence scores of benign samples, we randomly select 30 benign samples from the target class and generate a heatmap to display the distribution of average logits across n scaled models. As shown in Figure A11, the heatmaps for various attacks highlight a clear clustering phenomenon: scaled models consistently increase prediction confidence for a specific non-original label, thereby reducing confidence scores for the original labels of these benign samples. Moreover, these non-original labels correspond to categories that are more challenging to classify in CIFAR-10 dataset, typically having lower accuracy. For example, the category associated with Logit_3 is "Cat," which has an accuracy of less than 90%, noticeably below the dataset's average accuracy of approximately 94%.

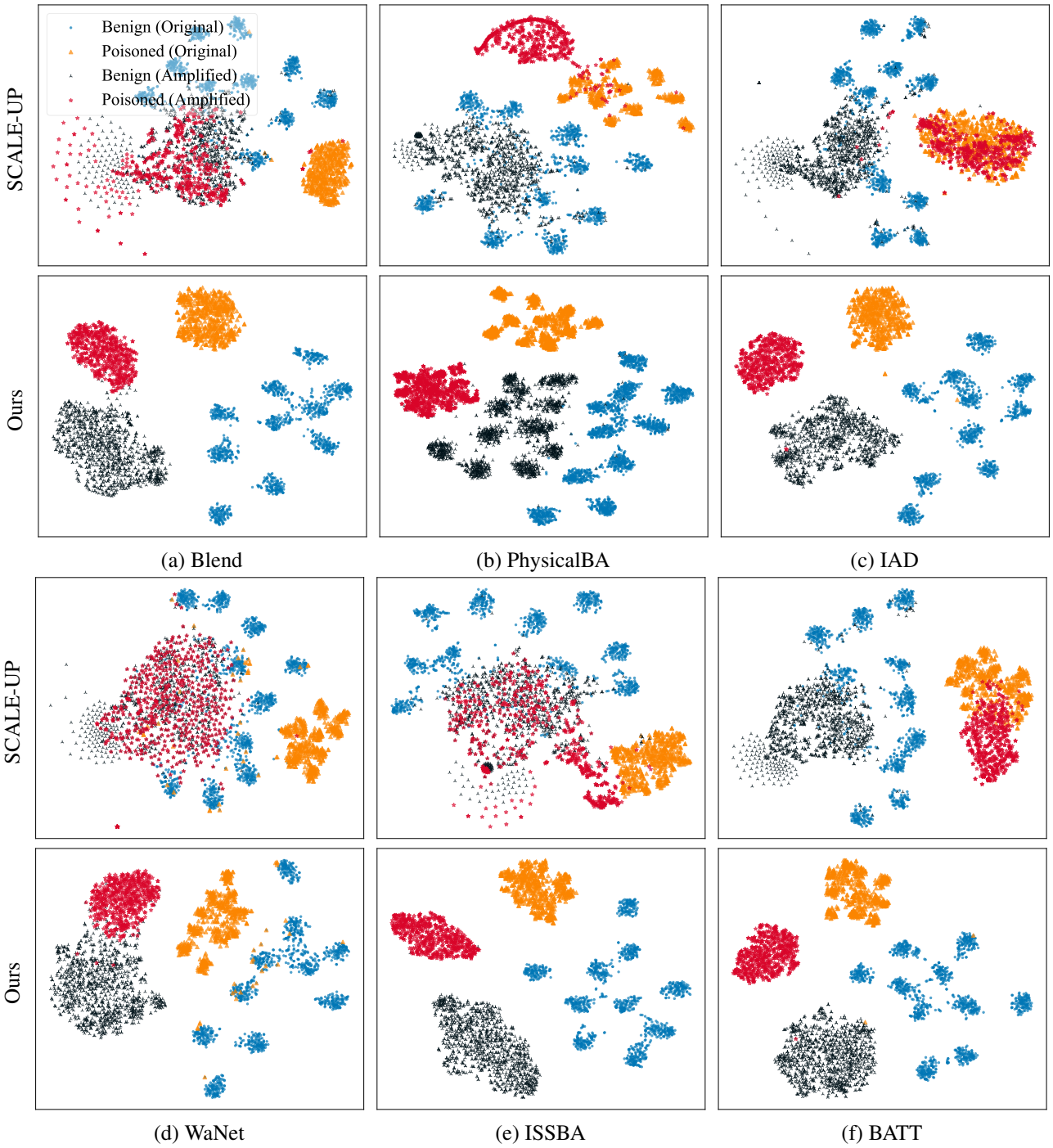


Figure A12. The t-SNE of feature representations of benign and poisoned samples on CIFAR-10 dataset against different backdoor attacks.

P. How Model Amplification Changes the Latent Representation

In this section, we provide a comprehensive set of t-SNE visualizations for all the attacks considered in our study. These visualizations show how the hidden layer features of benign and poisoned samples change under the modifications by SCALE-UP and our defense strategy. As indicated in Figure A12 and Figure A13, the amplification of pixel values by SCALE-UP results in a limited change within the feature space. In contrast, our defense achieves a more pronounced shift by modifying the model parameters, providing a more discernible differentiation between benign and poisoned samples. This is intuitively why our method achieves better performance in backdoor attack detection.

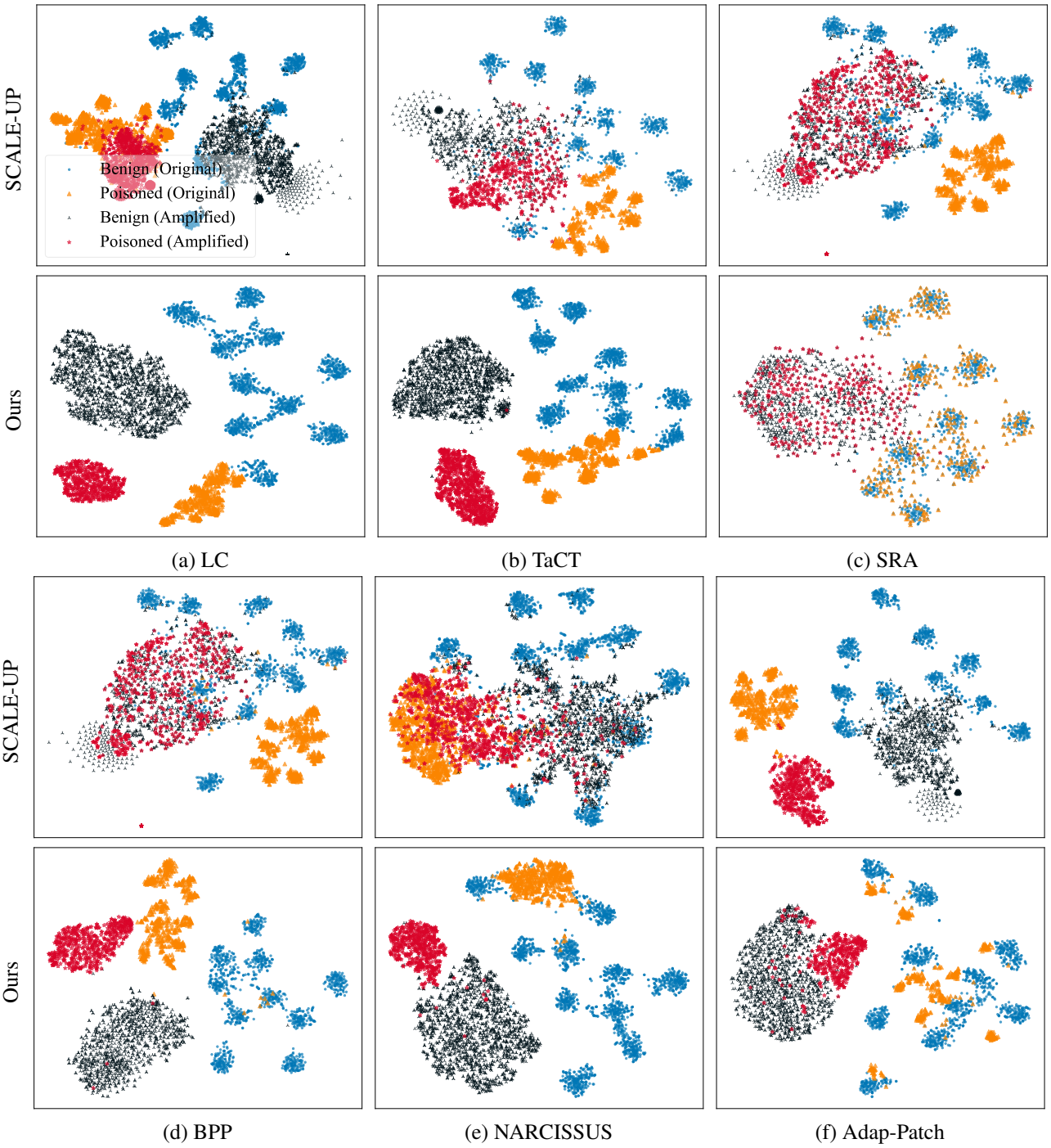


Figure A13. The t-SNE of feature representations of benign and poisoned samples on the CIFAR-10 dataset against other backdoor attacks (clean-label, specific-class, training-controlled, model-controlled, and adaptive attacks).

Q. The Extension to Training Set Purification

Q.1. Related Work

Training set purification aims to filter out potentially poisoned samples from a contaminated training set, ensuring that the model trained on the purified dataset is free from backdoors. Many existing studies assume that backdoored models will develop abnormal latent representations for poisoned samples, which are significantly different from those of benign samples,

Table A17. The performance (AUROC, TPR, FPR) of our defense on identifying the potential training poisoned samples.

Attacks →	BadNets			WaNet			BATT		
Defenses ↓	AUROC	TPR	FPR	AUROC	TPR	FPR	AUROC	TPR	FPR
MSPC	0.980	1.000	0.144	0.747	0.551	0.186	0.986	0.991	0.131
CD	0.980	0.895	0.052	0.710	0.303	0.121	0.767	0.403	0.117
Ours	1.000	1.000	0.066	0.998	1.000	0.081	0.994	1.000	0.079

Table A18. Effect of retraining models without poisoned samples identified by our defense.

Attacks	ASR	BA	# Removed Samples
BadNets	0.005	0.893	7985
WaNet	0.002	0.878	9119
BATT	0.009	0.860	8536

allowing for the identification of poisoned samples. Chen *et al.* (Chen et al., 2018) firstly observed that samples within the target class form two separate clusters in the feature space of the penultimate layer. They employed cluster analysis techniques, such as K-means, to segregate these clusters. Samples from the smaller cluster are classified as poisoned, based on the assumption that the number of poisoned samples is significantly lower than that of benign samples. Subsequent works generally utilized different cluster analysis methods, such as singular value decomposition (SVD) (Tran et al., 2018; Hayase et al., 2021), Gram matrix (Ma et al., 2022), K-Nearest-Neighbors (Peri et al., 2020), and feature decomposition (Tang et al., 2021), to detect poisoned samples. Another line of research focuses on identifying poisoned samples based on differentiating characteristics, such as the faster speed of model fitting (Li et al., 2021b), the presence of high-frequency artifacts (Zeng et al., 2021), and the sensitivity of poisoned samples to transformations (Chen et al., 2022).

More recently, Huang *et al.* (Huang et al., 2023) hypothesized that poisoned samples require less input information to be predicted correctly. They introduced the cognitive pattern signature technique, which distills a minimal pattern (given by a mask) for an input sample to retain its original prediction. This technique reveals that poisoned samples typically exhibit a significantly smaller L1 norm in the cognitive pattern compared to benign samples. Pan *et al.* (Pan et al., 2023) proposed a proactive training set purification method called ASSET, which maximizes the loss difference between poisoned and benign samples by optimizing opposite objectives on the base and poisoned sets. Pal *et al.* (Pal et al., 2024) presented intriguing observations regarding the limitations of SCALE-UP (Guo et al., 2023b), leading to the proposal of the masked scaled prediction consistency (MSPC) technique. This method selectively amplifies specific pixels in input samples, thereby more effectively exposing the prediction invariance of poisoned data under an input scaling factor.

Q.2. Comparing Our IBD-PSC with MSPC

MSPC (Pal et al., 2024) presents observations that are similar to ours regarding SCALE-UP. However, it is important to clarify that our findings on SCALE-UP constitute only a minor component of our research. While our study shares some similarities with the SCALE-UP framework, it significantly diverges by exploring parameter scaling, highlighting a substantial difference from MSPC. Additionally, our work focuses on different application scenarios compared to MSPC, which is primarily concerned with training set purification.

Our Method Requires Fewer Assumptions about Potential Adversaries. We explore scenarios where the users employ third-party models and need real-time detection of poisoned samples during the inference phase, similar to a firewall. This setup aligns with the framework proposed in SCALE-UP. Notably, we do not limit our adversaries to using poison-only attack methods, which is required by MSPC.

Differences in Detection Focus. Our defense operates during the inference stage, requiring the capability of real-time detection. In contrast, MSPC is less constrained in detection time as it operates during the data collection phase.

Q.3. Identifying and Filtering Potentially Poisoned Samples within Training set

Following the methodology in references (Pal et al., 2024; Huang et al., 2023; Pan et al., 2023), we first train a model on a potentially compromised training set and then apply our detection method to identify and filter potentially poisoned samples within that dataset. The detection performance, presented in Table A17, demonstrates the effectiveness of our method in filtering malicious training samples across various attacks, achieving a 100% TPR and nearly 100% AUROC

while maintaining an FPR close to 0%. Note that we reproduce MSPC using its open-source codes with default settings. However, it performs relatively poorly in defending against WaNet compared to the results reported in its original paper. We speculate this is because we test WaNet in noise mode, whereas MSPC is tested on the vanilla WaNet (as mentioned in their Appendix E). After removing suspected poisoned samples from the training set, we retrain the model on this purified training set to evaluate both its BA and ASR. We conduct experiments on the CIFAR-10 dataset against three representative attacks, and the results, presented in Table A18, show that the ASR scores of these retrained models are less than 0.5%, rendering these backdoor attacks ineffective.

R. Potential Limitations and Future Directions

In this section, we analyze the potential limitations and future directions of this work.

Firstly, our defense requires more memory and inference times than the standard model inference without any defense. Specifically, let M_s and M_d denote the memory (for loading models) required by the standard model inference and by that of our defense, respectively. Let T_s and T_d denote the inference time required by the standard model inference and by that of our defense. Assuming that we adopt n (*e.g.*, $n = 5$) parameter-amplified models for our defense. We have the following equation: $M_d \cdot T_d = n \times M_s \cdot T_s$. Accordingly, the users may need more GPUs to load all/some amplified models simultaneously to ensure efficiency or require more time for prediction by loading those models one by one when the memory is limited. In particular, the storage costs of our defense are similar to those without defense since we can easily obtain amplified models based on the standard one and, therefore, only need to save one model copy (*e.g.*, vanilla model). We will explore how to reduce those costs in our future work.

Secondly, our IBD-PSC requires a few local benign samples, although their number could be small (*e.g.*, 25, as shown in Figure A8). We will explore how to extend our method to the 'data-free' scenarios in our future works.

Thirdly, our method can only detect whether a suspicious testing image is malicious. Currently, our defense cannot recover the correct label of malicious samples or their trigger patterns. As such, the users can only mark and refuse to predict those samples. We will explore how to incorporate those additional functionalities in our future works.

Fourthly, our work currently focuses only on image classification tasks. We will explore its performance on other modalities (*e.g.*, text and audio) and tasks (*e.g.*, detection and tracking) in our future work.

S. Reproducibility Statement

We have provided detailed descriptions encompassing the datasets utilized, training and evaluation settings, along the computational resources involved. To facilitate the replication of our experimental results, the corresponding codes and model checkpoints have been provided in the supplementary materials.

T. Discussions about the Adopted Data

In this paper, all the samples we used are from publicly available datasets, including CIFAR-10, GTSRB, and ImageNet. It's worth noting that our defense method is implemented by modifying the pre-trained model parameters, without making any alterations to the input samples themselves. Therefore, our study doesn't raise any concerns regarding the privacy of human-related images within the dataset.