An Imperceptible Data Augmentation Based Blackbox Clean-Label Backdoor Attack on Deep Neural Networks

Chaohui Xu, *Graduate Student Member, IEEE*, Wenye Liu, *Member, IEEE*, Yue Zheng, *Member, IEEE*, Si Wang, and Chip-Hong Chang, *Fellow, IEEE*

***Abstract*— Deep neural networks (DNNs) have permeated into many diverse application domains, making them attractive targets of malicious attacks. DNNs are particularly susceptible to data poisoning attacks. Such attacks can be made more venomous and harder to detect by poisoning the training samples without changing their ground-truth labels. Despite its pragmatism, the clean-label requirement imposes a stiff restriction and strong conflict in simultaneous optimization of attack stealth, success rate, and utility of the poisoned model. Attempts to circumvent the pitfalls often lead to a high injection rate, ineffective embedded backdoors, unnatural triggers, low transferability, and/or poor robustness. In this paper, we overcome these constraints by amalgamating different data augmentation techniques for the backdoor trigger. The spatial intensities of the augmentation methods are iteratively adjusted by interpolating the clean sample and its augmented version according to their tolerance to perceptual loss and augmented feature saliency to target class activation. Our proposed attack is comprehensively evaluated on different network models and datasets. Compared with state-of-the-art clean-label backdoor attacks, it has lower injection rate, stealthier poisoned samples, higher attack success rate, and greater backdoor mitigation resistance while preserving high benign accuracy. Similar attack success rates are also demonstrated on the Intel Neural Compute Stick 2 edge AI device implementation of the poisoned model after weight-pruning and quantization.**

***Index Terms*— Clean-label backdoor attack, data augmentation, data poisoning, deep neural networks, edge AI.**

I. Introduction

DEEP neural networks (DNNs) have become a core technology today that drives leapfrog development of many smart applications including face recognition [1], self-driving [2], and healthcare [3]. Along with their thriving market successes come heightened security threats and attack surfaces. Deep learning models have been found to be notoriously susceptible to backdoor attack, as the data that fuels its predictive power is itself an exploitable vulnerability. Unlike hardware trojans, whose payloads and triggers are typically embedded through rogue insiders or third-party intellectual property integration, a backdoor can be secretly implanted into a DNN via training pipelines, including data collection, labeling and pre-processing, model training, etc.

Backdoor triggers in early works are mostly patch-based [4], [5], [6], which can be easily spotted by human eyes. Recent works [7], [8], [9] can poison training data with invisible triggers to evade visual inspection. Albeit visually imperceivable, the poisoned samples of these advanced backdoor attacks can still be easily detected by their dirty labels due to the mismatch between the label and the contextual and semantic information. Backdooring a DNN by poisoning a small subset of training data without changing their ground-truth labels is a very challenging task. Only a handful of clean-label backdoor attacks are reported in the literature [10], [11], [12], [13]. These attacks, while more pragmatic, are not without limitations. Existing clean-label backdoor attacks suffer from a low attack success rate. To compensate for the weak trigger, a relatively large injection rate is required, which increases the attack costs and reduces its stealth. Most clean-label backdoor attacks rely on a surrogate model to generate poisoned training samples, which assume white-box or gray-box knowledge about the model. Often, the attack performance is sensitive to the surrogate model architecture with poor transferability across models and domains. Moreover, existing backdoor attacks are evaluated only on software implementation of the model. With increasing data analytic and decision-making processes performed at the edge, it is imperative to also attest the attack success rates of poisoned networks implemented on edge AI accelerators.

In this paper, we proposed a novel clean-label backdoor attack utilizing data augmentation (DA) techniques as the trigger. It has overcome the limitations of our preliminary data-augmentation based backdoor attack that was primarily designed for the less challenging goal of dirty-label poisoning [14]. The poisoned sample is created by amalgamating three different DA techniques independently applied to the RGB channels of the clean sample. Our data poisoning and backdoor activation mechanism is unique in that the trigger

Manuscript received 6 April 2023; revised 3 July 2023; accepted 19 July 2023. Date of publication 4 August 2023; date of current version 18 December 2023. This work was supported in part by the National Research Foundation, Singapore; in part by the Cyber Security Agency of Singapore under its National Cybersecurity Research and Development Program/Cyber-Hardware Forensic and Assurance Evaluation Research and Development Program under Grant NRF2018NCR-NCR009-0001 and Grant CHFA-GC1-AW01; and in part by the Ministry of Education, Singapore, through the Academic Research Fund Tier 2 under Grant MOE-T2EP50220-0003. This article was recommended by Associate Editor P. Zhou. *(Corresponding author: Chip-Hong Chang.)*

Chaohui Xu, Wenye Liu, Yue Zheng, Si Wang, and Chip-Hong Chang are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: chaohui001@e.ntu.edu.sg; echchang@e.ntu.edu.sg).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSI.2023.3298802.

Digital Object Identifier 10.1109/TCSI.2023.3298802

is sample-specific, perceptual loss aware, target class activation sensitive, semantically consistent, and requires neither knowledge about model architecture nor transfer learning for backdoor embedding. Specifically, our major contributions can be summarized as follows:

- We propose a new clean-label backdoor attack that utilizes an amalgamation of DA techniques as backdoor triggers to inconspicuously poison the training data. The code is made available at https://github.com/Dshm212/adaptive-DA-attack.
- We propose an optimization algorithm to adaptively augment a small subset of training samples to maximize the backdoor activation success rate while minimizing the perceptual loss. It takes a lower than 5% injection rate of adaptively augmented clean-label samples to poison a model in black-box setting to achieve a higher than 90% target misclassification rate using merely the plain augmented trigger at inference time.
- Extensive experiments are conducted on four different networks and four different datasets to demonstrate its superior benign accuracy, attack success rate, stealth, generalizability, and robustness against state-of-the-art backdoor defenses.
- The attack is evaluated on a general-purpose computing platform and an Intel Neural Compute Stick 2 (NCS2) edge AI accelerator. Very high attack success rates were achieved on both platforms across models and datasets.

The rest of the paper is organized as follows. Related works are reviewed in Section II. A formal definition of the backdoor attack is provided in Section III. The proposed clean-label backdoor attack is elaborated in Section IV, followed by experimental results and analysis in Section V. The paper is concluded in Section VI.

II. Related Works

A. Dirty-Label Backdoor Attack

Data poisoning attacks date back to 2012 [15], [16], where a support vector machine classifier was misclassified by inserting carefully crafted attack samples into its training data. More sophisticated approaches were proposed later to compromise machine learning models by injecting adversarial attack points into the training set without prior knowledge about the learning models [17], [18]. These untargeted attacks significantly degrade the accuracy and lower the utility of the learning models. The trainer can be easily drawn into awareness of the attack once the training is completed or even during the training process.

BadNets [4] is a more covert but venomous data poisoning-based backdoor attack. The backdoor trigger is a malicious pattern placed at a fixed location of a small portion of training images with their labels modified to the target label. The poisoned images are mixed with the benign training samples. The trained victim model maintains a high prediction accuracy for benign images. However, when the same malicious pattern is added to any input image, it will be misclassified into the target label with a high probability. As the backdoor triggers are visually perceptible, and the modified labels of the poisoned samples are inconsistent with their content semantics, the poisoned samples can be easily identified and filtered out by screening the training dataset for apparently extraneous patterns or inconsistent labels.

To evade detection, Chen et al. [19] proposed to blend the trigger with benign images to keep the poisoned image stealthy. Techniques that follow this line of thought include using an adaptive perturbation mask that is derived from both the DNN model and the training set as an invisible backdoor trigger [9]; using a steganography technique to embed a pre-defined text into the training samples as the backdoor trigger [8]; and using warping transformations with limited amplitude (WaNet) to create poisoned images [7]. In fact, robust sample-specific backdoor triggers can also be generated using DNN-based steganography [20].

B. Clean-Label Backdoor Attack

Notwithstanding the imperceptible triggers, poisoned images of dirty-label backdoor attacks can be easily discovered by their semantically unrelated labels. Therefore, Turner et al. [10] first suggested a more practical clean-label backdoor attack without changing the ground-truth labels of poisoned training samples. In their label-consistent (LC) attack, the features of the images to be poisoned in the target class are first weakened by adversarial perturbations or latent space interpolation using generative adversarial networks (GANs) to prevent the salient benign features from being adequately learned. A patch trigger similar to [4] is then placed on a fixed location of the poisoned images to make it a dominant feature for the target class. In [21], inconspicuous sinusoidal stripes instead of the localized patch are used as a distinguishing backdoor trigger for clean-label backdoor attack. A different clean-label attack was recently proposed by Liu et al. [22], which modeled the reflection phenomenon to generate natural-looking poisoned samples.

In Feature Collision (FC) [12], clean-label poisoned training samples are created by solving a bilevel optimization problem. To enhance the transferability, the targeted image is surrounded with poisoned images in the feature space by Convex Polytope (CP) [13]. Both FC and CP attacks have very low generality since only a specific test instance can activate the desired misclassification. Hidden Trigger Backdoor Attack (HTBA) [11] can imperceptibly embed triggers into clean training samples, and the implanted backdoor can be triggered by unseen poisoned samples. The backdoors are embedded by transfer learning in these three attacks, which require the pre-trained network to be retrained with the poisoned samples. This requirement severely limits the attack scenario and feasibility. They are inapplicable if the models are randomly initialized and trained from scratch.

The limitations and deficiencies of existing clean-label backdoor attacks motivate us to develop a clean-label backdoor attack that possesses the following desiderata: it can be performed in a black-box setting with no knowledge about the victim model's structure and weights, the backdoor features can be learned by the victim model without requiring transfer learning, and the backdoor can be activated with unseen

TABLE I
COMPARISON OF CLEAN-LABEL BACKDOOR ATTACKS BASED ON WHETHER THE POISONED TRAINING SAMPLES ARE CREATED UNDER THE **W**HITE-BOX OR **B**LACK-BOX SETTING ASSUMPTIONS, THE BACKDOOR IS EMBEDDED INTO THE VICTIM MODEL BY TRAINING THE MODEL FROM **S**CRATCH OR BY **T**RANSFER LEARNING, AND WHETHER THE BACKDOOR CAN BE ACTIVATED WITH UNSEEN TEST INSTANCES

Ref	Attack	Setting	Training	Unseen sample triggerable
[10]	LC	B	S	✓
[21]	SIG	B	S	✓
[22]	ReFool	B	S	✓
[12]	FC	W	T	×
[13]	CP	B	T	×
[11]	HTBA	W	T	✓
ours	DA-based	B	S	✓

poisoned test instances. Table I summarizes the attributes of clean-label attacks for image classification tasks.

C. Backdoor Defense

1) Backdoor Detection: One straightforward approach to defend against poisoning-based backdoor attacks is to purify the training or test dataset. Chen et al. [23] demonstrated that activations of the last hidden layer can be used to explain model decisions and spot abnormal training points. The victim model is fed with all the training samples of each candidate class to collect the activations of the last hidden layer. These activations are then divided into two groups by k-means clustering and the Silhouette score is calculated. A high Silhouette score implies that the candidate class is likely to be poisoned. To identify if a test sample at inference time contains a trigger, strong intentional perturbation (STRIP) [24] superimposes a suspected test sample with a set of clean samples from different classes before they are input into the victim model. The entropy of their predictions determines if the test sample is poisoned.

Another approach is to identify if a model is poisoned. Neural Cleanse [25] is the first approach that detects backdoor attacks by finding an outlier from the reverse-engineered triggers of every target label. For each target class, a minimal size trigger that can misclassify clean samples from all other classes to the target class is created. The smallest outlier trigger with anomaly index above a threshold is identified as the real trigger and its corresponding target class is the true target class. DeepInspect [26] improved upon Neural Cleanse by reconstructing triggers for multiple suspected labels without any clean data and in one run, thus relaxing the reliance on prior knowledge of the defender and reducing the computational cost. In SentiNet [27], Grad-CAM [28] is exploited to explain the model prediction and reveal the backdoor trigger. By observing the behaviors of untrusted models on carefully designed samples, many surrogate benign and backdoored models are generated to train a meta-classifier to predict if an untrusted model is attacked [29]. Artificial Brain Stimulation (ABS) [30] scans the neurons of the suspected model to determine if it is backdoored.

2) Backdoor Mitigation: In [31], the backdoor effect was eliminated and the poisoned model is repaired by pruning dormant neurons. The method records the response of all neurons to a set of clean inputs and iteratively prunes neurons from the least important to the most important. As neuron pruning can result in an unacceptable loss of accuracy, fine-tuning is required to restore the accuracy of the pruned model. Based on the observation that infected neurons exhibit strong sensitivity to adversarial neuron perturbations, Adversarial Neuron Pruning (ANP) [32] is proposed to identify and prune these vulnerable neurons to purify the model.

Backdoor effects can also be mitigated during training time. In [33], a pre-trained autoencoder is employed to remove agnostic backdoor triggers while preserving the benign features. Febrrus [34] is a two-stage preprocessing method for sanitizing poisoned samples. The first step identifies contiguous sub-regions that have significant contributions to model decisions through Grad-CAM [28] and replaces these regions with solid-color masks. In the second step, these masked regions are reconstructed with a GAN to obtain purified training samples. The backdoor attack success rate can also be brought down without compromising prediction accuracy by applying some strong data transformations, such as MixUp, CutMix and MaxUp, during the training stage [35]. Anti-Backdoor Learning (ABL) [36] uses a gradient ascent based learning to isolate suspicious training samples with extremely small loss at the early training epochs, and then unlearns their backdoor features at the later training stage.

III. PRELIMINARIES

A. Formalization of Backdoor Attacks

A computer vision DNN model can be denoted by $f_\theta : \mathbb{X} \to \mathbb{Y}$, where $\mathbb{X}$ represents an image domain, $\mathbb{Y} = \{y_1, y_2, \cdots, y_k\}$ is a set of k classes, and θ denotes the parameters that a model learns from the training dataset $\mathcal{D}_{tr} = \{(x_i, y_i)|x_i \in \mathbb{X}, y_i \in \mathbb{Y}, i = 1, 2, \cdots N_{tr}\}$. Let (x_i', y_t) be a poisoned training sample of (x_i, y_i) obtained by adding a specific adversarial feature to the clean image x_i, where x_i' is a poisoned image and $y_t \in \mathbb{Y}$ is the target class label. The adversarial feature is the backdoor trigger. A poisoned training dataset, $\mathcal{D}_{tr}^p$ can be formed by replacing a subset of clean training samples $\mathcal{D}_{tr\_s}^c \subset \mathcal{D}_{tr}$, by its poisoned training samples, $\mathcal{D}_{tr\_s}^p$. Then, $\mathcal{D}_{tr\_r} \subset \mathcal{D}_{tr}$ is a set of untainted clean images and $\mathcal{D}_{tr\_r} = \mathcal{D}_{tr}^p \setminus \mathcal{D}_{tr\_s}^p$. The injection rate $\gamma = m/N_{tr}$ denotes the fraction of poisoned samples in the training set, where $m = \left|\mathcal{D}_{tr\_s}^p\right|$.

The benign training with $\mathcal{D}_{tr}$ can be regarded as a single-level optimization problem. The objective is to obtain an accurate model f_θ. The optimization seeks to solve the following problem during training:

$$\min_\theta \mathcal{L}(\mathcal{D}_{tr}, f_\theta) = \sum_{i=1}^{N_{tr}} l(x_i, y_i, f_\theta), \tag{1}$$

where $l(\cdot)$ is the loss function (e.g., the cross-entropy), and $(x_i, y_i) \in \mathcal{D}_{tr}$.

Backdoor training, on the other hand, is a bi-level optimization problem. A backdoor attack is successful if the victim model $f_{\theta'}$ behaves normally on benign inputs but misclassifies any inputs embedded with the same trigger to the target label

y_t with high probability, i.e.,

$$\begin{cases} f_{\theta'}(x_i) = y_i, \\ f_{\theta'}(x_i') = y_t. \end{cases} \tag{2}$$

The optimization objective in the training phase is:

$$\min_{\theta'} \mathcal{L}(\mathcal{D}_{tr\_r} \bigcup \mathcal{D}_{tr\_s}^{p}, f_{\theta'}) = \sum_{i=1}^{N_{tr}-m} l(x_i, y_i, f_{\theta'}) + \sum_{j=1}^{m} l(x_j', y_t, f_{\theta'}). \tag{3}$$

The first term of (3) corresponds to the benign training. It makes the poisoned model learn to behave normally as if it is a backdoor-free model with the clean set D_{tr_r}. The second term forces $f_{\theta'}$ to learn the association of the backdoor trigger with the target label y_t. It causes the trained model to output the target label when it encounters the same trigger.

Under the clean-label setting, the attacker can only plant backdoor triggers into a small subset of training samples whose ground-truth label is exactly the same as the target label y_t. Therefore, the clean and poison samples are represented as (x_i, y_t) and (x_i', y_t), respectively.

B. Threat Model

1) Attack Scenarios: There are two commonly conceived avenues for which a poisoning-based backdoor attack can be mounted, which are through the third-party dataset and machine learning as a service (MLaaS). The main difference between them lies in the adversary's knowledge about the training pipeline. In the first scenario, the developers may train their models on publicly available datasets, or outsource or crowdsource the training data collection to save time and effort. This provides an opportunity for an attacker to supply a poisoned training dataset to the developer or infiltrate poisoned samples into a training dataset for a model. In the MLaaS scenario, the clients may train their models on untrusted cloud platforms [8]. A malicious platform can easily modify the benign datasets of MLaaS clients with full knowledge of the model architecture and full control of the training process. The attackers can also distribute pre-trained backdoored models to the clients for further fine-tuning, or directly provide cloud DNN inference services through application programming interfaces. Though not as broadly mentioned, there is also a possibility that the attacker discreetly hacked into the training database to sneak or replace some training data.

Attack through the third-party dataset is more pragmatic but also more challenging. This is also the main target scenario of our proposed attack. To make the scenario more viable, the following assumptions are made, which add to the challenge of a successful attack.

2) Attackers' Knowledge and Capabilities: Our attack is designed for the black-box scenario. We assume that the attacker knows part of the training samples and is capable of poisoning a small fraction of them but cannot change the data label. The attacker is allowed to train a surrogate model on clean training data to extract latent features from the intermediate layers. This is performed with no knowledge about other components of the training pipeline (e.g., model architecture, training schedule, training loss, optimization algorithm, etc.), nor accessibility to the training procedure or the well-trained model. Hence, it is not feasible for the attacker to embed backdoors by directly manipulating the training procedure or model weights. In the inference stage, we assume that the attacker can only submit poisoned test samples to the victim model but is not able to access the prediction outputs.

3) Attackers' Goals: The attacker aims to plant a hidden backdoor into a victim model unnoticed with the following goals.

Effectiveness: The most important goal is to maximize the attack success rate (ASR). The poisoned inputs are expected to be misclassified to the target label, regardless of their source labels. The ASR is computed by:

$$\begin{cases} \text{ASR} = \dfrac{1}{N_{ts}} \displaystyle\sum_{i=1}^{N_{ts}} I\left(f_{\theta'}(x_i'), y_t\right), \\ \text{s.t.} \quad (x_i, y_i) \in \mathcal{D}_{ts}, \;\; y_i \neq y_t, \end{cases} \tag{4}$$

where $\mathcal{D}_{ts}$ denotes the clean test set and $I(\cdot)$ is the attack activation function. $I(\cdot)$ is set to 1 if $f_{\theta'}(x_i') = y_t$ and 0 otherwise.

As the attacker may not have control over the entire training set, the injection rate should be kept low.

Stealth: The poisoned model is expected to perform well on benign test images to keep the victim model useful and avoid arousing attention. Therefore, the benign accuracy (BA) should be comparable to a clean model with negligible degradation if any. On the other hand, the backdoor triggers should be imperceptible and free from unnatural distortions to evade visual inspection. In this paper, four popular full reference image quality assessment (IQA) metrics are used to objectively evaluate the perceptual loss induced by the backdoor trigger. They are Learned Perceptual Image Patch Similarity (LPIPS) [37], Peak Signal-to-Noise Ratio (PSNR) [38], Gradient Magnitude Similarity Deviation (GMSD) [39] and Structural Similarity Index (SSIM) [40].

Robustness: The backdoor of a poisoned model and the trigger of a poisoned test image should not be detectable or removable by existing backdoor attack detection and mitigation methods. Meantime, the backdoor of the victim model should still be successfully triggered upon application of known defense techniques.

IV. Proposed Clean-Label Backdoor Attack

The overall pipeline of our proposed method is shown in Fig. 1. Three different DA techniques are separately applied to the RGB channels to hide the malicious features in the image to be poisoned. To increase the triggering sensitivity of DA features without introducing visually perceivable or unnatural distortion into the host image of the trigger, we propose an iterative optimization algorithm for generating a sample-specific interpolation mask and its inverse to adaptively redistribute the spatial intensity of DA. The clean host and its plainly augmented image are interpolated with the generated masks to maximally utilize its backdoor feature hosting capacity.

A. Amalgamated Data Augmentation Trigger

The trigger of our proposed clean-label backdoor attack is a unique set of DA features. The reasons are: 1) DA can be

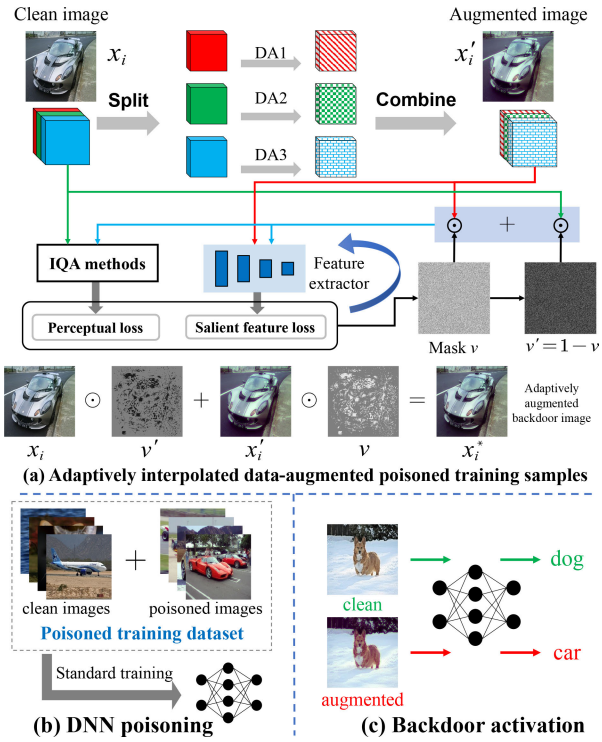


Fig. 1. Overall pipeline of the proposed data augmentation-based backdoor attack.

used to generate a large variety of realistic and natural-looking poisoned images that are visually indistinguishable from the source images drawn from real scenes. 2) DA has been commonly used to increase the size and diversity of the training dataset. Efficient DA techniques are available for image classification including basic image manipulations and machine learning approaches, such as adversarial training, neural style transfer, and GAN-based techniques [41]. 3) DA is content-dependent. The perturbations induced by the same DA on different samples are different, which render backdoor defense methods like [25] that assume content-independent trigger ineffective. 4) DA is good for the adaptation of DNN, which acts positively to improve the ASR of particularly clean-label backdoors. As the decision boundaries of a complex DNN with lower bias are more easily overstrained by the source distribution of the training dataset, DA can help to reduce the variance of complex models and improve domain adaptation. Many techniques, such as dropout [42] and batch normalization [43], have been attempted to overcome the overfitting and domain shifts of DNN models but they are not as effective as DA in optimizing the bias-variance tradeoff. Such space can be exploited for keeping the prediction accuracy of untainted data while enhancing the imperceptibility of the trigger and the success rate of backdoor activation.

To increase the robustness and the backdoor feature hosting capacity, we propose to embed a composite backdoor trigger into a clean image by amalgamating three DA techniques. Each DA technique is applied independently onto a separate color channel of the image to decouple different DA effects. This will reduce the unnatural distortions due to mixed perturbations in one channel. The following three simple steps are used to poison a clean training sample (x_i, y_t) into (x_i', y_t).

- Split the color image x_i into red (R), green (G), and blue (B) channels. The corresponding single-channel images are, $x_{i(r)}$, $x_{i(g)}$ and $x_{i(b)}$.
- Select three DA techniques, $\mathcal{A}_1(\cdot)$, $\mathcal{A}_2(\cdot)$, and $\mathcal{A}_3(\cdot)$, and apply them onto $x_{i(r)}$, $x_{i(g)}$ and $x_{i(b)}$, independently.
- Combine the three independently data-augmented channels, $\mathcal{A}_1(x_{i(r)})$, $\mathcal{A}_2(x_{i(g)})$, and $\mathcal{A}_3(x_{i(b)})$, into a single poisoned color image x_i'.

The poisoned image generation can be succinctly expressed as follows:

$$\begin{aligned} x_i' &= \mathcal{F}(x_i, \mathcal{A}_1(\cdot), \mathcal{A}_2(\cdot), \mathcal{A}_3(\cdot)) \\ &= [\mathcal{A}_1(x_{i(r)}), \mathcal{A}_2(x_{i(g)})\mathcal{A}_3(x_{i(b)})], \end{aligned} \tag{5}$$

where $\mathcal{F}(\cdot)$ denotes the poisoning function.

As the perceptual loss and feature strength of DA transformation varies with the color channel, image resolution and dataset on which they are applied, an adaptive trade-off optimization algorithm is proposed in Sec. IV-B. Our experiments show that the proposed adaptive augmentation provides good stealth and trigger efficiency for various DA transformations. We noted certain DA may not blend well with other DA across the color channels or have limited room for trade-off optimization between visual imperceptibility and distinctive features at pixel levels. DAs such as rotation, translation, shearing and rescaling, which cause inconsistent geometric distortion across channels, are to be avoided. Also, as human eyes are more sensitive to green color, pixel-level transformations, such as histogram equalization and brightness adjustment, as opposed to spatial-level transformations like sharpening and blurring, are preferred for the green channel.

For convenience, we arbitrarily select Gaussian blurring (GB), brightness and contrast adjustment (BC), and kernel smoothing (KS) as three efficient DA transformations for backdoor trigger embedding and provide an ablation study with other DA combinations in Sec. V-E. The extent (intensity) of these DAs can be controlled by their corresponding user-defined scaling parameters as indicated in the functions below:

$$\begin{cases} \mathrm{GB}(x_i) = x_i \otimes \mathcal{G}(\sigma), \\ \mathrm{BC}(x_i) = \alpha * x_i + \beta * 255, \\ \mathrm{KS}(x_i) = x_i \otimes \mathcal{K}(\delta), \end{cases} \tag{6}$$

where x_i is the original clean image and $\otimes$ denotes the convolutional operation. $\mathcal{G}(\sigma)$ is the 2D Gaussian convolutional kernel with standard deviation σ in both X and Y directions. The size of the Gaussian kernel is automatically computed from σ by the OpenCV library. α and β are parameters that control contrast and brightness, respectively. $\mathcal{K}(\delta)$ is the smoothing kernel with a size of $\delta \times \delta$, and with each parameter in the convolutional kernel set to $1/\delta^2$.

To keep the modifications contributed due to these augmentations imperceptible, the parameters σ, α, β, and δ are tuned to minimize the perceptual loss while keeping the amalgamated DA features sufficiently strong.

B. Asymmetric and Capacity-Aware Backdoor Trigger

Most of the existing backdoor attacks utilize symmetric backdoor triggers, which means that the strength of the mali-

cious patterns embedded in the training and test samples are exactly the same. Barni et al. [21] measured the attack performance of their sinusoidal stripe trigger under the asymmetric setting and demonstrated that the ASR can be increased if the intensity of the stripes on the test samples is slightly stronger than that on the training samples. Such results are not surprising since the embedded backdoor can be more easily activated if a stronger backdoor feature is presented to the victim model than what it learned during the training process. The One-to-N attack in [5] improves the ASR by poisoning the training samples with asymmetric trigger intensities. The method does not consider the difference in perceptual tolerance of the samples to be poisoned. As a result, some samples that have lower perceptual tolerance to the trigger patterns are not stealthy and can be easily spotted.

We also apply asymmetric backdoor triggers on the training and test samples to enhance the victim model's sensitivity to the malicious feature but in a perceptual loss-aware manner. Specifically, the clean test samples are uniformly poisoned with a **plain** composite DA trigger, whereas the training samples are **adaptively** poisoned with the same type of DA trigger but with sample-specific and spatially non-uniform intensities. Since the perceptual tolerance of DA features (henceforth refers to as backdoor capacity) varies among images and across regions of an image, we propose an iterative algorithm to adapt the DA trigger strength at the pixel level to the backdoor capacity of each training image to be poisoned.

Since the intensity of each plain DA trigger applied on the test image is sample agnostic, the parameters of its three DA methods are empirically determined to achieve a high ASR and generically low perceptual loss. Given a specific clean RGB training image x_i and its plainly augmented version x_i', an adaptively augmented poisoned image x_i^* can be produced by a convex combination of x_i and x_i' to adjust the DA intensity of x_i' according to the backdoor capacity of x_i as follows:

$$x_i^* = x_i' \odot v + x_i \odot (1 - v), \tag{7}$$

where $\odot$ denotes an element-wise multiplication operation, and $v \in \mathbb{R}^{h \times w \times 1}$ is a two-dimensional interpolation mask of the same size as x_i and x_i'. All elements in v range from 0 to 1.

As v controls the interpolation of pixel intensity between the two images. Its generation can be considered as a trade-off optimization between the **effectiveness** and **stealth** of the backdoor trigger. The goal is to maximize the learnability of the embedded trigger by the victim model and the imperceptibility of the trigger. The coefficients of the interpolation mask should be designed such that high-strength DA transformations are preserved on regions of x_i that possess high perceptual tolerance against the applied DA features and DA features are reduced in regions that have low perceptual tolerance to their presence. At the same time, DA features in regions of x_i' that are more effective in activating the target output class should be weighted more than regions with inert DA features.

To assess the relative likelihood between two samples in activating a specific output class, a surrogate model f_s trained on the clean training set $\mathcal{D}_{tr}$ is used to measure the distance between the adaptively augmented image x_i^* and the plainly augmented image x_i' in the latent feature space. In this work, AlexNet [44] is adopted as the surrogate model f_s since it is simple and the training cost is trivial. We select the last ReLU layer as the intermediate hidden layer to extract the latent features for the calculation of the following penalty function:

$$\mathcal{L}_e = \sum_{i=1}^{m} \frac{\| f_s(x_i^*) - f_s(x_i') \|}{\| f_s(x_i') \|} \tag{8}$$

where $\| f_s(x_i^*) \|$ and $\| f_s(x_i') \|$ are the intermediate representations of x_i^* and x_i', respectively.

All latent feature distances are normalized to prevent any one sample from dominating the objective loss. The penalty function $\mathcal{L}_e$ pushes the adaptively augmented image x_i^* towards the plainly augmented image x_i' if they are close in the latent space. A corner case is $\mathcal{L}_e = 0$. When all elements of the interpolation mask are 1, x_i^* and x_i' are exactly the same, and they have the same latent features.

To ensure that each adaptively augmented image looks natural and the perturbations it introduced are visually imperceptible, an objective function to assess its perceptual similarity with the clean image x_i is introduced. The penalty function is made up of three IQAs and is expressed as follows:

$$\mathcal{L}_s = \sum_{i=1}^{m} (\mathcal{L}_{lpips}^{(i)} + \mathcal{L}_{psnr}^{(i)} + \mathcal{L}_{gmsd}^{(i)}). \tag{9}$$

The three loss terms in (9) can be calculated by:

$$\begin{cases} \mathcal{L}_{lpips}^{(i)} = \text{ReLU}(lpips(x_i^*, x_i) - \lambda_1), \\ \mathcal{L}_{psnr}^{(i)} = \text{ReLU}(\lambda_3 - psnr(x_i^*, x_i)), \\ \mathcal{L}_{gmsd}^{(i)} = \text{ReLU}(gmsd(x_i^*, x_i) - \lambda_2), \end{cases} \tag{10}$$

where $lpips(\cdot)$, $psnr(\cdot)$ and $gmsd(\cdot)$ denote the IQA functions, LPIPS, PSNR and GMSD, respectively between x_i^* and x_i. $\lambda_{1,2,3}$ are the pre-defined thresholds, and ReLU [45] is a commonly used activation function that sets the output to zero when the input is negative and passes non-negative input as is to the output.

By combining the penalty functions of feature loss and perceptual loss, the overall objective function is given by:

$$\min_{\{v_i\}_{i=1}^m} \mathcal{L}_{total} = \mathcal{L}_e + \mathcal{L}_s. \tag{11}$$

The algorithm for generating the adaptively interpolated training images is presented in Algorithm 1, where the interpolation mask is updated iteratively. Due to the GPU memory size constraint, interpolation masks are calculated and optimized in batches.

To prevent the backdoor features from being too weak to be learned, the lower bound lb of all elements in the interpolation mask is set to 0.5 to ensure that the plainly augmented image x_i' contributes to at least half of the adaptively augmented image x_i^*. Additionally, the upper bound of all elements in the interpolation mask is set to 0.9 to guarantee that the intensities of the poisoned training images are always lower than those of the poisoned test images even in the corner case. This is to enhance the ASR by asymmetric training and test trigger strength. It should be noted that we only apply Algorithm 1 to generate adaptively augmented poisoned training samples. For test samples, the triggers are embedded by applying plain DA transformations.

Algorithm 1 Adaptively Augmented Training Images Generation

Input: Clean training images $\{x_i\}_{i=1}^m$, poisoning function $\mathcal{F}(\cdot)$, surrogate model f_s, IQA thresholds $\lambda_{1,2,3}$, mask elements range $[lb, ub]$, number of epochs E, the batch size B, learning rate η.

Output: Adaptively interpolated training images $\{x_i^*\}_{i=1}^m$.

for *each batch* $\{x_j\}_{j=1}^B$ *from* $\{x_i\}_{i=1}^m$ **do**
  Generate plainly augmented images $\{x_j'\}_{j=1}^B$ by (5)
  Initialize masks $\{v_j\}_{j=1}^B$
  for *epoch=1, 2, ⋯, E* **do**
    Generate adaptively augmented images $\{x_j^*\}_{j=1}^B$ by (7)
    Compute loss:
      $\mathcal{L}_{total} = \mathcal{L}_e + \mathcal{L}_s$
    Calculate gradients $\{g_j\}_{j=1}^B$ w.r.t. $\{v_j\}_{j=1}^B$ by backward propagation
    for *j=1, 2, ⋯, B* **do**
      Update interpolation mask:
        $v_j = v_j - \eta \cdot g_j$
      Clip elements in v_j to the defined range:
        $v_j = \text{Clip}(v_j, lb, ub)$
  Generate optimized backdoor images $\{x_j^*\}_{j=1}^B$ by (7)
Include all batches of optimized backdoor images into $\{x_i^*\}_{i=1}^m$
return $\{x_i^*\}_{i=1}^m$

V. Experimental Results and Discussions

A. Experimental Setup

1) Datasets: We use four image classification datasets, CIFAR-10 [46], ImageNet [47], Food-101 [48] and CelebA [49], for our evaluation. The first three datasets are used for object recognition, and the last dataset is used for face recognition.

CIFAR-10 contains 60K 32×32 color images uniformly distributed in 10 classes. Due to the large number of classes in ImageNet, it is a common practice to arbitrarily select 10 classes of images to speed up the evaluation without loss of generality. For the same reason, 10 classes of images are arbitrarily selected from Food-101 dataset. CelebA is a large-scale face attributes dataset that consists of over 200K images and each has 40 attributes. Following the configuration suggested in [50], we concatenate the three most balanced attributes (Heavy Makeup, Mouth Slightly Open, Smiling) to create an eight-class subset. The details of these datasets are listed in Table II.

2) Network Structures: Four popular classifiers are employed as the victim models. They are ResNet18 [51], EfficientNet-B0 [52], VGG16 [53] and DenseNet121 [54]. ResNet18 is a common network used by state-of-the-art backdoor attacks for performance evaluation. Hence it is used as the reference network for comparison with existing clean-label attack methods and the default network for robustness evaluation. Nevertheless, we will demonstrate the generalizability of our attack by comparing its BAs and ASRs on the other three model architectures with those on ResNet18. AlexNet is used as the surrogate model f_s for all the experiments.

TABLE II
Datasets Used for the Experiments

Dataset	# Labels	Input Size	# Training / Test Images
CIFAR-10	10	32×32	50000 / 10000
ImageNet-10	10	224×224	12000 / 1000
Food-10	10	224×224	8000 / 2000
CelebA	8	64×64	48000 / 7775

TABLE III
Parameters of Data Augmentation Methods

Dataset	σ	α	β	δ
CIFAR-10	0.5	0.3	0.36	3
ImageNet-10	2.5	0.3	0.36	9
Food-10	2.5	0.3	0.36	9
CelebA	0.75	0.3	0.36	3

3) Implementation Platforms: We evaluate our backdoor attacks on victim models implemented on a general-purpose computing platform and an edge AI device. The former consists of an AMD Ryzen 3960X CPU and two NVIDIA GeForce RTX 3090 GPUs. The latter is an Intel Neural Computing Stick 2 (NCS2) VPU. The difference is that pre-trained models are always pruned and quantized before deploying onto edge devices. Before a pre-trained model can be implemented onto NCS2, a fraction of its least useful weights in the convolutional layers is removed by a global unstructured pruning operation and the remaining weights are quantized from 32-bit to 16-bit floating-point representation.

4) Other Settings: The injection rate is set to 4% for CIFAR-10, ImageNet-10, and Food-10 datasets. Unlike dirty-label poisoning whereby the labels of all classes other than the target class can be altered to poison any samples, only samples from the target class can be poisoned in clean-label poisoning. This means that for the same injection rate of 4%, 40% of images in the target class in each of these three datasets are poisoned since the training samples are uniformly distributed to 10 classes. For the CelebA dataset, we only poison 10% of the images in the target class. Therefore, the injection rate is 1.25%. The default batch size used by Algorithm 1 to generate the poisoned training images is 32. For each sample to be poisoned, 200 iterations are run to produce the final interpolation mask. All DNNs are trained with a Stochastic Gradient Descent optimizer [55] for 120 epochs. The initial learning rate is set to 0.01 and reduced by a factor of 0.1 after every 40 training epochs. During the training process, we also apply several commonly used DA methods to increase the generalizability of the model and reduce overfitting, including random horizontal flip, random rotation, and random erasure. Table III lists the scaling parameters of the plain DA applied

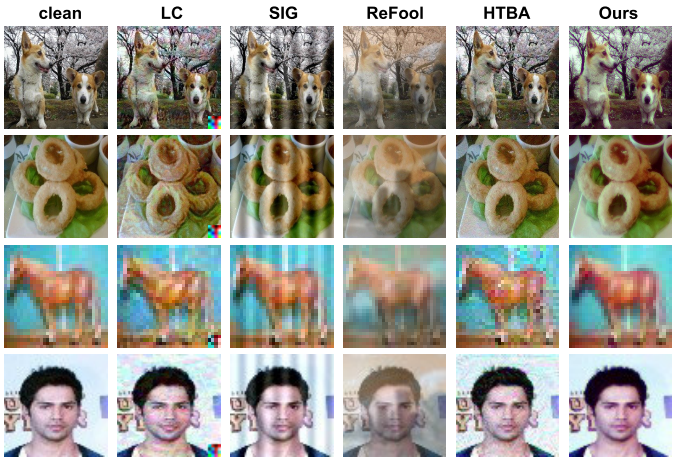


Fig. 2. Visual comparison of images poisoned by the proposed and existing clean-label backdoor attack methods.

on each color channel to cater for different image resolutions of the four datasets.

B. Stealth of DA-Based Trigger

We compare the test images embedded with our plain DA backdoor trigger against the same images embedded with backdoor triggers generated by four existing clean-label backdoor attacks, namely LC [4], SIG [21], ReFool [22] and HTBA [11]. For LC, the default size of the square trigger is 1/64 of the original clean image. The SIG trigger is defined as $v(i, j) = \Delta sin(2\pi jf/m)$. We set $\Delta = 30$ and $f = 6$ as recommended in [21]. For ReFool, we randomly select images from the PascalVOC dataset [56] as the reflection images. The adversarial perturbations generated by HTBA are restricted by ℓ_∞-norm with $\varepsilon = 32$, and we conduct this attack in the transfer learning scenario.

Four different poisoned test images of these methods for the four datasets are shown in Fig. 2 for visual comparison. The poisoned test images of ours in the last column are obtained by plain augmentation. This is to demonstrate that even without the adaptive augmentation, our plainly augmented images are already stealthier than those poisoned by other attack methods. From Fig. 2, the images poisoned by LC and SIG are visually distinguishable from the corresponding clean images due to their image-agnostic and visible backdoor triggers (the square pattern at the bottom right corner and the vertical stripes). In addition, LC uses adversarial perturbations to dilute the original salient features of the target class, which causes the poisoned images to appear unnaturally distorted. ReFool tries to use reflection effects to generate realistic images. However, the blurring effects are visually perceivable. In contrast, our DA-based trigger is sample-specific and the resulting poisoned image is visually imperceptible. Table IV provides a quantitative comparison of the stealth of images poisoned by these attack methods on the four datasets based on four objective IQA metrics. The best IQA score in each row is printed **bold**. Our plainly augmented backdoor trigger outperforms the other three attacks in almost all cases. For those cases where it does not score the best, it scores the next best except LPIPS for CIFAR-10. HTBA scores exceptionally well on LPIPS for the low-resolution CIFAR-10 images while the other three methods have comparable scores for this case.

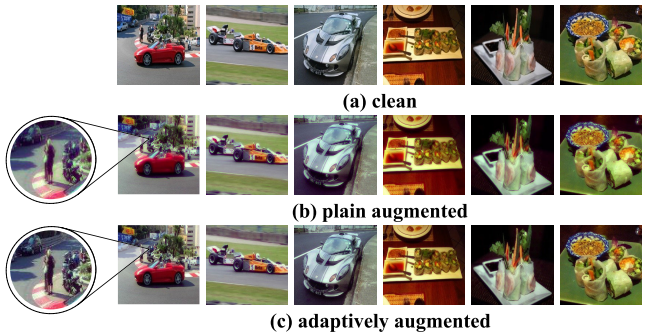


Fig. 3. Visual comparison between clean, plainly augmented, and adaptively augmented training images.

TABLE IV
COMPARISON OF IQA SCORES BETWEEN IMAGES POISONED BY THE PROPOSED AND EXISTING CLEAN-LABEL BACKDOOR ATTACK METHODS

Dataset	Perceptual Loss	Attack Method				
		LC	SIG	ReFool	HTBA	Ours-plain
ImageNet-10	LPIPS	0.335	0.261	0.243	0.228	**0.212**
	GMSD	0.149	0.213	0.119	0.057	**0.048**
	PSNR	21.858	18.503	15.343	**28.752**	25.303
	SSIM	0.705	0.741	0.755	0.788	**0.944**
Food-10	LPIPS	0.304	0.224	0.218	0.214	**0.197**
	GMSD	0.158	0.206	0.119	0.052	**0.050**
	PSNR	21.347	18.617	14.924	**28.861**	25.678
	SSIM	0.704	0.724	0.745	0.769	**0.944**
CIFAR-10	LPIPS	0.045	0.035	0.072	**0.018**	0.048
	GMSD	0.061	0.110	0.163	0.028	**0.014**
	PSNR	22.138	25.204	15.923	25.195	**25.796**
	SSIM	0.883	0.815	0.662	0.899	**0.981**
CelebA	LPIPS	0.088	0.070	0.172	0.054	**0.052**
	GMSD	0.103	0.248	0.174	0.050	**0.014**
	PSNR	22.045	22.058	14.366	25.713	**27.404**
	SSIM	0.858	0.700	0.625	0.858	**0.973**

We further show that the stealth of plainly augmented images is not better than the corresponding adaptively augmented images generated by Algorithm 1. In Fig. 3, the top row displays the clean training images, the middle row shows its plainly augmented versions, and the bottom row shows the corresponding adaptively augmented images. At the first glance, the subtle differences between the middle and bottom rows of images are not visually detectable. A closer observation by zooming in the images, the contrast between them can be seen by the small perturbations appeared in the plainly augmented images, particularly at the boundaries of complex textures. Despite they are decoupled into channels, the perturbations introduced by the three selected DA techniques may still interfere with each other upon amalgamation in high-frequency regions where small gradient changes are more readily detectable by human eyes. These regions have low visual tolerance to intensity alteration. Algorithm 1 takes into account the visual tolerance to pixel intensity change of a given image for DA strength reduction. Hence, the adaptively augmented images look natural even at these boundaries. Table V shows the IQA scores of plainly augmented, randomly augmented, and adaptively augmented poisoned images. A randomly augmented image is obtained by a convex combination of a clean training image x_i and its plainly

TABLE V

COMPARISON OF IQA SCORES AMONG PLAINLY AUGMENTED, RANDOMLY AUGMENTED, AND ADAPTIVELY AUGMENTED TRAINING IMAGES

Dataset	Perceptual Loss	Attack Strategy		
		plain	random	adaptive
ImageNet-10	LPIPS	0.179	0.107	**0.077**
	GMSD	0.065	0.038	**0.029**
	PSNR	24.189	27.253	**28.931**
	SSIM	0.927	0.960	**0.972**
Food-10	LPIPS	0.183	0.111	**0.081**
	GMSD	0.050	0.030	**0.023**
	PSNR	25.933	28.852	**30.190**
	SSIM	0.947	0.970	**0.977**
CIFAR-10	LPIPS	0.048	0.025	**0.021**
	GMSD	0.017	**0.009**	**0.009**
	PSNR	25.903	28.971	**29.398**
	SSIM	0.981	**0.990**	**0.990**
CelebA	LPIPS	0.056	0.029	**0.018**
	GMSD	0.013	0.007	**0.006**
	PSNR	27.093	30.159	**31.307**
	SSIM	0.974	0.985	**0.987**

TABLE VI

BAS (%) AND ASRS (%) OF DIFFERENT CLEAN-LABEL BACKDOOR ATTACK METHODS ON THE FOUR DATASETS

Dataset		ImageNet-10		Food-10		CIFAR-10		CelebA	
		BA	ASR	BA	ASR	BA	ASR	BA	ASR
Attack	None	87.80	0.78	77.25	2.72	94.72	0.67	72.62	3.63
	LC	87.90	71.48	78.32	90.37	94.45	93.83	72.87	**99.89**
	SIG	88.07	74.93	77.55	92.20	94.23	93.52	73.00	86.84
	ReFool	87.27	65.41	76.93	79.83	93.91	64.87	72.71	90.19
	HTBA[1]	97.77	63.50	81.98	50.33	79.89	67.87	75.36	72.43
	plain	87.57	69.74	77.07	90.30	94.17	93.42	73.29	92.34
	random	87.47	85.30	76.53	98.31	94.12	96.31	72.58	99.79
	adaptive	88.30	**91.89**	77.37	**98.84**	94.20	**98.06**	72.99	99.47

[1] BAs of HTBA are obviously different from others because this attack is performed in the transfer learning scenario.

augmented image x_i', with a fixed interpolation mask v where the elements of the mask are randomly selected from the range $[lb, ub]$, i.e., $v_{ij} = \text{Random}[lb, ub], 0 \leq i < h, 0 \leq j < w$, where h and w are the height and width, respectively of the interpolation mask v. The best IQA score of each row of Table V is printed **bold**. The results show that the adaptively augmented images always have the best score, irrespective of the datasets.

C. Attack Performance

The BA and ASR of our proposed attack are evaluated and compared with reported results of other clean-label backdoor attacks in Table VI for the four datasets. The row "None" refers to the clean pre-trained model. Its accuracy for each dataset is used as the baseline for assessing the BA degradation of the poisoned models. The ASR of the clean model for each dataset is obtained by applying the same poisoned test image of our method. Since the clean model does not have any backdoor embedded, its ASR is the false positive prediction rate of the target class, which is very low. The last three rows refer to our proposed DA attacks.

The BAs of all poisoned models show that the accuracy degradations are negligible compared to the clean models for the same datasets, implying that normal functionality is not affected by the backdoors embedded into these victim models.

TABLE VII

PERFORMANCE OF THE PROPOSED DA-BASED CLEAN-LABEL BACKDOOR ATTACK ACROSS NETWORKS

DataSet		Model			
		R-18	E-B0	V-16	D-121
ImageNet-10	BA (%)	88.30 (87.80)	87.90 (88.33)	86.53 (86.83)	86.87 (88.37)
	ASR (%)	91.89	95.87	96.96	93.33
Food-10	BA (%)	77.37 (77.25)	78.85 (80.02)	67.60 (68.75)	76.37 (76.73)
	ASR (%)	98.84	97.83	96.52	98.52
CIFAR-10	BA (%)	94.20 (94.72)	85.41 (85.63)	93.41 (94.01)	95.43 (95.44)
	ASR (%)	98.06	95.34	99.20	99.31
CelebA	BA (%)	72.99 (72.62)	73.92 (73.99)	71.92 (72.32)	72.14 (72.22)
	ASR (%)	99.47	98.21	99.49	99.93

Our proposed DA-based attack has the highest ASRs on ImageNet-10, Food-10, and CIFAR-10 when the models are poisoned with adaptively augmented training images. LC has the highest ASR on CelebA, but only 0.42% higher than our adaptively augmented poisoned model. Even though HTBA is a white-box attack, its ASRs, ranging from 50.33% to 72.43%, are much lower than ours for all four datasets. In fact, the ASRs of all other clean-label backdoor attacks are substantially lower than ours for the high-resolution ImageNet-10 dataset. We also analyze the ASRs of our proposed attack using three different DA strategies for poisoning the training images. The ASRs of the two constant DA-poisoned models are relatively lower than that of the adaptive DA-poisoned model. The plainly augmented poisoned model achieves only 69.74% ASR on ImageNet-10. Although the random DA poisoning backdoor can be better learned and activated than the plainly augmented backdoor, it is still not as effective as the adaptive poisoning backdoor.

We also test if our proposed adaptively augmented clean-label backdoor can be successfully activated on the poisoned model deployed on NCS2. Due to the weight-pruning and quantization operations performed on the poisoned model prior to its physical implementation, we assume that a maximum of 4% drop in the BA of the full-precision clean model is allowed for DNN models deployed on edge AI devices. We also measure the BAs and ASRs of the poisoned models with different weight-pruning ratios ranging from 10% to 90%. As shown in Fig. 4, the BA and ASR are barely affected when the pruning ratio is lower than 50%. However, as the pruning ratio increases further, the BA decreases sharply and falls below the 4% BA degradation threshold, while the ASR remains relatively constant. For instance, the BA of the poisoned model for CelebA implemented on NCS2 drops by 7.76% when 70% of the least significant weights of the original poisoned model are pruned, but the embedded backdoor can still be activated with an ASR of 99.80%. Although increasing the pruning ratio to 80% or 90% can substantially reduce the ASR, the deployed models will fail to behave normally. This attested that our proposed attack can withstand typical weight-pruning and quantization operations for edge AI deployment.

To demonstrate that our proposed adaptive DA-based clean-label attack is network independent, we also evaluate it on three other models, EfficientNet-B0 (E-B0), VGG16 (V-16), and DenseNet121 (D-121), and compare the results against

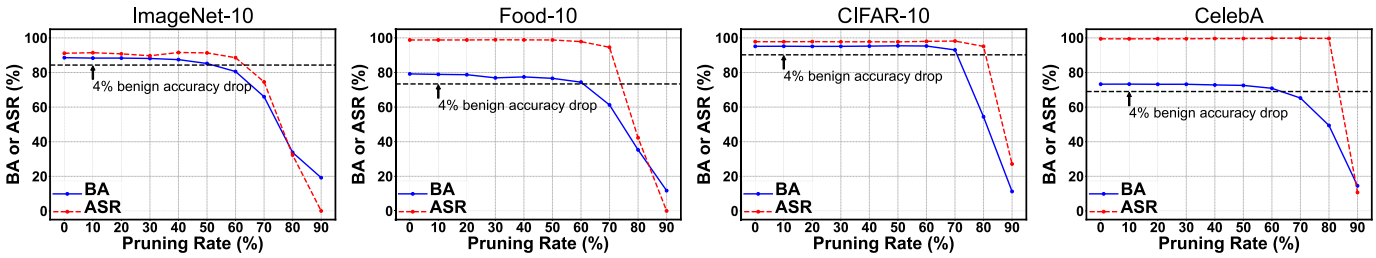


Fig. 4. Attack performance of our proposed clean-label attack on NCS2 edge AI device with different weight pruning rates.

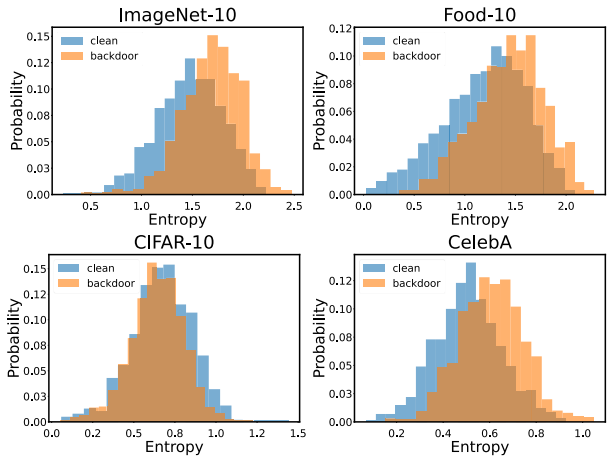


Fig. 5. Entropy of the proposed attack generated by STRIP.

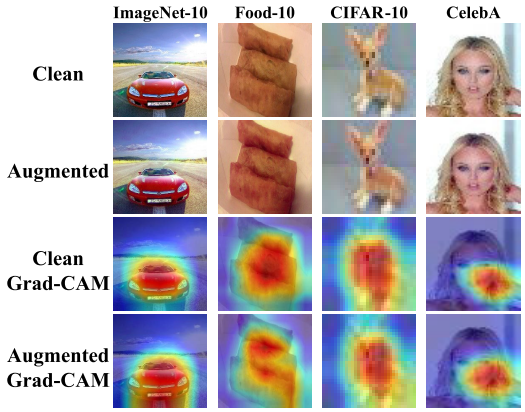


Fig. 6. Grad-CAM visualization of clean and adaptively augmented images generated by the poisoned model.

those presented earlier for ResNet18 (R-18) in Table VII. The bracketed numbers in the BA(%) rows refer to the accuracies of the corresponding clean models. The results show that the accuracy degradation due to backdoor embedding is trivial. Our proposed attack can still achieve the same high ASR on all these network models, though their architectures are very different from AlexNet, which is employed as the surrogate model f_s. These results indicate that our DA-based attack has good transferability and poses a more severe threat than previously reported clean-label backdoor attacks.

D. Resistance to Backdoor Defenses

We evaluate the robustness of the proposed attack against eight backdoor defenses, which include four backdoor detection methods: STRIP [24], SentiNet [27], Neural Cleanse [25], and Activation Clustering [23], and four backdoor elimination methods: fine-pruning [31], Anti-Backdoor Learning [36], sanitizing training [35], and Adversarial Neuron Pruning [32].

1) STRIP: Fig. 5 shows the detection results of applying STRIP to the clean and poisoned test images for the network poisoned by our proposed attack on the four datasets. It shows that the clean and backdoor images have similar and non-separable entropy distributions. Since our DA-based trigger is sample-specific and content-adaptive, superimposing different image content onto a poisoned image will annihilate its trigger and lead to equally random prediction as superimposing different classes of images onto a clean image.

2) SentiNet: In Fig. 6, the first row consists of clean images and the second row contains poisoned images created by our proposed attack. We separately feed the clean and poisoned images to the victim model and compare the network behaviors. It is observed that the attention heatmaps generated by the victim model for both clean and poisoned images are very similar. In particular, the model consistently focuses on the central areas of the input, regardless of whether it has been poisoned, which indicates that Grad-CAM fails to identify the trigger regions of images poisoned by our attack. The content-adaptive and sample-specific backdoor trigger of our attack violates the attributes of trigger detectable by SentiNet, which assume that the target class of a poisoned network is activated by the presence of small, localized, and constant perturbations.

TABLE VIII

THE DETECTION RESULTS OF THE NEURAL CLEANSE METHOD AGAINST THE PROPOSED ATTACK ON THE FOUR DATASETS

Dataset	ImageNet-10	Food-10	CIFAR-10	CelebA	Threshold
Anomaly Index	0.3296	1.4924	0.9154	0.6534	2.0

3) Neural Cleanse: Table VIII shows the anomaly index of the reverse-engineered trigger found by Neural Cleanse for the actual target class of the network poisoned by our proposed attack for each of the four training datasets. It shows that the reverse-engineered trigger for the target class obtained by Neural Cleanse is lower than the anomaly threshold of 2, which means that our proposed attack can evade Neural Cleanse detection. This is because Neural Cleanse is designed to detect small and fixed triggers. It is impossible for it to reverse engineer a non-localized, adaptive and sample-specific backdoor trigger.

4) Activation Clustering: Fig. 7 shows the Silhouette scores of all classes of CIFAR-10 obtained by Activation Clustering of the network poisoned by our proposed method. The actual poisoned class is "dog", which has a Silhouette score of 0.36.

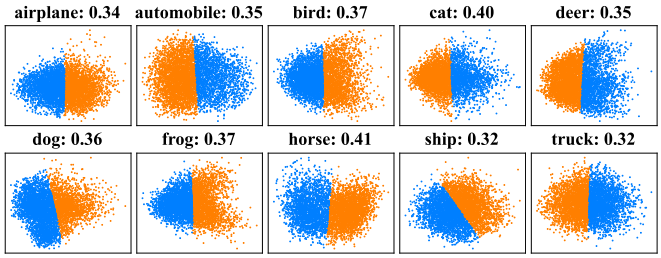


Fig. 7. The results of Activation Clustering on CIFAR-10 dataset. Class and Silhouette score are given at the top of each subfigure.

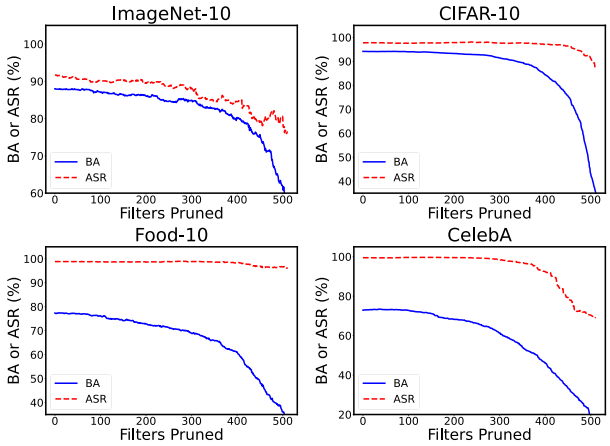


Fig. 8. BA and ASR of the network poisoned by the proposed attack versus the number of pruned filters under fine-pruning defense for ImageNet-10, CIFAR-10, Food-10, and CelebA.

TABLE IX

PERFORMANCE OF THE PROPOSED ATTACK TRAINED WITHOUT/WITH THE ANTI-BACKDOOR LEARNING SCHEME. THE DEVIATION INDICATES THE CHANGES IN BA/ASR COMPARED TO THE BASELINE

DataSet	BA (%)			ASR (%)		
	None	ABL	Deviation	None	ABL	Deviation
ImageNet-10	88.30	66.90	21.40	91.89	69.22	22.67
Food-10	77.37	66.00	11.37	98.84	96.28	2.56
CIFAR-10	94.20	73.72	20.48	98.06	99.87	-1.81
CelebA	72.99	71.95	1.04	99.47	99.96	-0.49

This is the fifth largest score out of the ten scores. Therefore, the Activation Clustering method fails to detect our attack for CIFAR-10. Similar results can be obtained by applying Activation Clustering on the models trained by the other three poisoned datasets.

5) Fine-Pruning: Fig. 8 shows the BA and ASR of the network poisoned by our proposed attack versus the number of filters pruned for four different datasets. The results show that the BA decreases far more rapidly with the number of pruned filters than the ASR for all the datasets. This means that a pruning point that can successfully mute the backdoor with a high probability before the network fails to function correctly does not exist. Hence, the backdoor embedded by our attack cannot be debilitated by fine-pruning.

6) Anti-Backdoor Learning: The BA and ASR of our poisoned models trained without/with ABL are presented in Table IX. ABL has negligible impact on the ASR of our DA-based backdoor for all but ImageNet datasets. For ImageNet, ABL has reduced the ASR of our method by 22.67%. It should be noted that ABL has also substantially reduced the BA of our poisoned models by 10-20% for all datasets except CelebA. The sample-specific backdoor trigger of our clean-label attack has deeply coupled with the benign features, causing ABL to miss the poisoned samples in early training epochs. Without altering the class label of poison samples, our clean-label attack also confuses ABL in relating the isolated benign features to their correct labels at the later training stage, resulting in an unacceptably high loss of utility.

TABLE X

RESISTANCE TO SANITIZING TRAINING DEFENSE

DataSet		Method			
		baseline	MixUp	CutMix	MaxUp
ImageNet-10	BA (%)	88.30	84.23	83.97	85.80
	ASR (%)	91.89	93.67	95.63	96.00
Food-10	BA (%)	77.37	69.37	70.80	69.97
	ASR (%)	98.84	98.54	99.02	99.76
CIFAR-10	BA (%)	94.20	94.17	93.55	94.17
	ASR (%)	98.06	99.61	99.67	99.54
CelebA	BA (%)	72.99	75.32	75.41	75.51
	ASR (%)	99.47	99.94	100.00	99.99

TABLE XI

RESISTANCE TO ADVERSARIAL NEURON PRUNING DEFENSE. THE DEVIATION INDICATES THE CHANGES IN BA/ASR COMPARED TO THE BASELINE

DataSet	BA (%)			ASR (%)		
	None	ANP	Deviation	None	ANP	Deviation
ImageNet-10	88.30	71.40	16.90	91.89	86.78	5.11
Food-10	77.37	67.70	9.67	98.84	85.00	13.84
CIFAR-10	94.20	92.15	2.05	98.06	86.79	11.27
CelebA	72.99	57.83	15.16	99.47	91.90	7.57

7) Sanitizing Training: Table X shows the BA and ASR of the poisoned network after applying each of the three DA-based sanitizing training, MixUp, CutMix and MaxUp, to purify the training samples poisoned by our proposed adaptively augmented DA trigger. The results show that the backdoor embedded in the victim models of all datasets can still be successfully triggered even after the poisoned models have undergone sanitized training.

8) Adversarial Neuron Pruning: We use the configuration suggested in [32] to implement ANP. The BA and ASR of our poisoned model purified by ANP are presented in Table XI. The results demonstrate that our attack can withstand ANP by maintaining a high ASR of at least 85% across all datasets. The ANP purification also incurs non-trivial BA degradation for all but CIFAR-10 datasets.

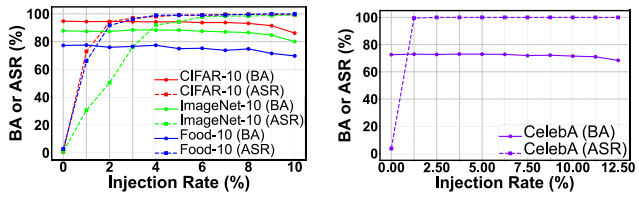
E. Ablation Study

1) Effect of DA Combinations: We evaluate the attack performance of six other DA transformations. We first replace each of GB, BC and KS with ToSepia (TS), contrast limited adaptive histogram equalization (CLAHE) and glass blurring (Glass), respectively for Food-10, and sharpening (SP), gamma transformation (GT), and median blurring (MB), respectively for CIFAR-10. We then completely replace the DA combination in all three channels with the new combination. As shown in Table XII, all new combinations of DA methods attain comparably good perceptual quality, utility and attack performance.

TABLE XII

ATTACK PERFORMANCE OF DIFFERENT DA TRANSFORMATIONS ON THE FOOD-10 AND CIFAR-10 DATASETS

Dataset	DA Combination	Perceptual Loss				Attack Performance	
		LPIPS	PSNR	GMSD	SSIM	BA (%)	ASR (%)
Food-10	GB+BC+KS	0.081	30.190	0.023	0.977	77.37	98.84
	TS+BC+KS	0.062	28.434	0.010	0.984	77.05	96.87
	GB+CLAHE+KS	0.081	30.494	0.023	0.984	76.91	98.65
	GB+BC+Glass	0.076	30.638	0.021	0.977	77.07	97.33
	TS+CLAHE+Glass	0.073	28.718	0.011	0.987	76.80	96.02
CIFAR-10	GB+BC+KS	0.021	29.398	0.009	0.990	94.20	98.06
	SP+BC+KS	0.013	28.573	0.007	0.985	94.42	98.97
	GB+GT+KS	0.015	30.260	0.004	0.991	94.33	97.89
	GB+BC+MB	0.020	29.346	0.008	0.990	94.20	97.72
	SP+GT+MB	0.013	28.878	0.003	0.995	94.11	98.36



(a) (b)

Fig. 9. BAs and ASRs of the proposed attack with different γ for the datasets (a) CIFAR-10, ImageNet-10 and Food-10, and (b) CelebA.

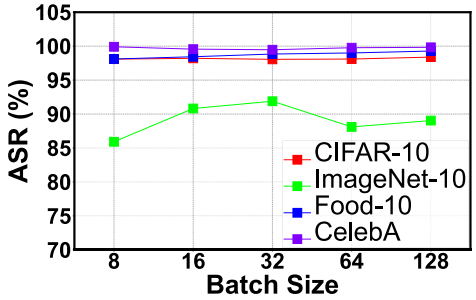


Fig. 10. ASR of the proposed attack with different batch sizes to generate adaptively augmented images.

2) Effect of Injection Rate γ: The default injection rate is set to 1.25% for CelebA and 4% for the other three 10-class datasets. Additional tests are conducted to evaluate the performance of our attack with different injection rates, and the results are presented in Fig. 9. For the three 10-class datasets, we gradually increase the injection rate from 0% to 10% with a step size of 1%. As shown in Fig. 9(a), the backdoor features embedded into the victim model are too weak to be triggered successfully with a relatively low injection rate. When γ is set to only 1%, the ASRs of ImageNet-10, Food-10, and CIFAR-10 are 30.56%, 66.00%, and 72.87%, respectively. As the injection rate increases, the ASRs rise rapidly but the BAs for the same datasets do not drop as fast. We find a 4% injection rate provides a good ASR-BA tradeoff on these three 10-class datasets. As for CelebA, the ASR is much higher even if $\gamma = 1.25\%$. This is likely because the CelebA dataset represents a more challenging training task to achieve good BA, leading to the original target class features being easily overwhelmed by the backdoor features during the training process and hence a higher ASR with a lower γ.

3) Effect of Batch Size Used for the Adaptively Augmented Images Generation: In Algorithm 1, the default batch size for creating adaptively augmented images is 32. We also evaluate the BA and ASR of our attack by varying the batch size from 8 to 128 to generate the poisoned images for the four datasets, as shown in Fig. 10. The results indicate that the impact of batch size on BA and ASR is negligible when it is 16 and above. Therefore, the batch size can be kept at around 16 or 32 without further optimization.

TABLE XIII

ATTACK PERFORMANCE OF THE PROPOSED ATTACK ON IMAGENET-10 DATASET WITH DIFFERENT SURROGATE MODELS

Victim Model		Surrogate Model		
		AlexNet	MobileNet	ShuffleNet
R-18	BA (%)	88.30	87.67	88.00
	ASR (%)	91.89	91.26	91.11
E-B0	BA (%)	87.90	88.97	87.87
	ASR (%)	95.78	97.15	97.48
V-16	BA (%)	86.53	86.33	85.87
	ASR (%)	96.96	96.04	95.96
D-121	BA (%)	86.87	87.17	87.30
	ASR (%)	93.33	94.04	93.60

4) Effect of Surrogate Model Architecture: Table XIII presents the BAs and ASRs of our proposed attack on different victim models trained on ImageNet-10 using different surrogate models, AlexNet, MobileNet, and ShuffleNet. The results show that our attack can achieve similar good BA and ASR with different surrogate models. This further corroborates the superior transferability of our attack in comparison to previous works. Previous works like FC [15] and HTBA [11] utilize the surrogate model to generate imperceptible perturbations as backdoor features, making the attack performance highly dependent on the similarity between the surrogate and victim models' architectures. In contrast, the surrogate model in our proposed attack is used mainly to determine the similarity in the latent space between a plainly augmented image and an adaptively augmented image that differs only in the DA intensities in certain regions of the two images. The relative differences in the latent features between two similarly augmented images do not vary much by using different model architectures for their extraction. Hence, the same high ASR can still be achieved even if the victim and surrogate model architectures are different.

VI. CONCLUSION

In this paper, we propose a novel clean-label backdoor attack that leverages efficient DA techniques to generate inconspicuous backdoor triggers. The proposed attack is validated on four image classification datasets and four deep learning network models. It achieves high ASRs without compromising the prediction accuracy of normal inputs for all evaluated models and datasets. Our attack outperforms existing clean-label backdoor attacks in stealth and effectiveness. The poisoned images are free from visually detectable unnatural distortions and artifacts, and have better IQA scores. Our adaptively augmented training data poisoning substantially increases the sensitivity of backdoor activation with the plainly augmented trigger in the inference stage. The equally successful attack on the NCS2 edge AI device demonstrates that the embedded backdoor is not affected by weight pruning and quantization

operations. Our attack is also proven to be robust and can withstand several widely used and powerful backdoor mitigation countermeasures.

ACKNOWLEDGMENT

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore, Cyber Security Agency and Ministry of Education of Singapore.

REFERENCES

[1] V. Ghenescu, R. E. Mihaescu, S.-V. Carata, M. T. Ghenescu, E. Barnoviciu, and M. Chindea, "Face detection and recognition based on general purpose DNN object detector," in *Proc. Int. Symp. Electron. Telecommun. (ISETC)*, Timişoara, Romania, Nov. 2018, pp. 1–4.

[2] T.-D. Do, M.-T. Duong, Q.-V. Dang, and M.-H. Le, "Real-time self-driving car navigation using deep neural network," in *Proc. 4th Int. Conf. Green Technol. Sustain. Develop. (GTSD)*, Ho Chi Minh City, Vietnam, Nov. 2018, pp. 7–12.

[3] K. Y. Ngiam and I. W. Khor, "Big data and machine learning algorithms for health-care delivery," *Lancet Oncol.*, vol. 20, no. 5, pp. e262–e273, May 2019.

[4] T. Gu, K. Liu, B. Dolan-Gavitt, and S. Garg, "BadNets: Evaluating backdooring attacks on deep neural networks," *IEEE Access*, vol. 7, pp. 47230–47244, 2019.

[5] M. Xue, C. He, J. Wang, and W. Liu, "One-to-N & N-to-one: Two advanced backdoor attacks against deep learning models," *IEEE Trans. Depend. Sec. Comput.*, vol. 19, no. 3, pp. 1562–1578, May 2022.

[6] Y. Liu et al., "Trojaning attack on neural networks," in *Proc. 25th Annu. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, San Diego, CA, USA, Feb. 2018, pp. 1–16.

[7] T. A. Nguyen and A. T. Tran, "WaNet—Imperceptible warping-based backdoor attack," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Apr. 2020, pp. 1–16.

[8] S. Li, M. Xue, B. Z. H. Zhao, H. Zhu, and X. Zhang, "Invisible backdoor attacks on deep neural networks via steganography and regularization," *IEEE Trans. Depend. Sec. Comput.*, vol. 18, no. 5, pp. 2088–2105, Sep./Oct. 2020.

[9] H. Zhong, C. Liao, A. C. Squicciarini, S. Zhu, and D. Miller, "Backdoor embedding in convolutional neural network models via invisible perturbation," in *Proc. 10th ACM Conf. Data Appl. Secur. Privacy*, Mar. 2020, pp. 97–108.

[10] A. Turner, D. Tsipras, and A. Madry, "Label-consistent backdoor attacks," 2019, *arXiv:1912.02771*.

[11] A. Saha, A. Subramanya, and H. Pirsiavash, "Hidden trigger backdoor attacks," in *Proc. 34th AAAI Conf. Artif. Intell.*, vol. 34, no. 7, New York, NY, USA, Apr. 2020, pp. 11957–11965.

[12] A. Shafahi et al., "Poison frogs! Targeted clean-label poisoning attacks on neural networks," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, QC, Canada, Dec. 2018, pp. 6106–6116.

[13] C. Zhu et al., "Transferable clean-label poisoning attacks on deep neural nets," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Long Beach, CA, USA, Jun. 2019, pp. 7614–7623.

[14] C. Xu, W. Liu, Y. Zheng, S. Wang, and C.-H. Chang, "Inconspicuous data augmentation based backdoor attack on deep neural networks," in *Proc. IEEE 35th Int. Syst.-on-Chip Conf. (SOCC)*, Belfast, U.K., Sep. 2022, pp. 1–6.

[15] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," 2012, *arXiv:1206.6389*.

[16] H. Xiao, H. Xiao, and C. Eckert, "Adversarial label flips attack on support vector machines," in *Proc. 20th Eur. Conf. Artif. Intell. (ECAI)*, Montpellier, France, Aug. 2012, pp. 870–875.

[17] M. Zhao, B. An, W. Gao, and T. Zhang, "Efficient label contamination attacks against black-box learning models," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Melbourne, VIC, Australia, Aug. 2017, pp. 3945–3951.

[18] H. Xiao, B. Biggio, B. Nelson, H. Xiao, C. Eckert, and F. Roli, "Support vector machines under adversarial label contamination," *Neurocomputing*, vol. 160, pp. 53–62, Jul. 2015.

[19] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, "Targeted backdoor attacks on deep learning systems using data poisoning," 2017, *arXiv:1712.05526*.

[20] Y. Li, Y. Li, B. Wu, L. Li, R. He, and S. Lyu, "Invisible backdoor attack with sample-specific triggers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 16463–16472.

[21] M. Barni, K. Kallas, and B. Tondi, "A new backdoor attack in CNNs by training set corruption without label poisoning," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Taiwan, Sep. 2019, pp. 101–105.

[22] Y. Liu, X. Ma, J. Bailey, and F. Lu, "Reflection backdoor: A natural backdoor attack on deep neural networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Online, Aug. 2020, pp. 182–199.

[23] B. Chen et al., "Detecting backdoor attacks on deep neural networks by activation clustering," Nov. 2018, *arXiv:1811.03728*.

[24] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "STRIP: A defence against trojan attacks on deep neural networks," in *Proc. 35th Annu. Comput. Secur. Appl. Conf.*, Austin, TX, USA, Dec. 2019, pp. 113–125.

[25] B. Wang et al., "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *Proc. IEEE Symp. Secur. Privacy (SP)*, San Francisco, CA, USA, May 2019, pp. 707–723.

[26] H. Chen, C. Fu, J. Zhao, and F. Koushanfar, "DeepInspect: A black-box trojan detection and mitigation framework for deep neural networks," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2019, pp. 4658–4664.

[27] E. Chou, F. Tramèr, and G. Pellegrino, "SentiNet: Detecting localized universal attacks against deep learning systems," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2020, pp. 48–54.

[28] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 618–626.

[29] X. Xu, Q. Wang, H. Li, N. Borisov, C. A. Gunter, and B. Li, "Detecting AI trojans using meta neural analysis," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2021, pp. 103–120.

[30] Y. Liu, W.-C. Lee, G. Tao, S. Ma, Y. Aafer, and X. Zhang, "ABS: Scanning neural networks for back-doors by artificial brain stimulation," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, London, U.K., Nov. 2019, pp. 1265–1282.

[31] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *Proc. Int. Symp. Res. Attacks, Intrusions, Defenses (RAID)*, Heraklion, Greece, Sep. 2018, pp. 273–294.

[32] D. Wu and Y. Wang, "Adversarial neuron pruning purifies backdoored deep models," in *Proc. 35th Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, vol. 34, 2021, pp. 16913–16925.

[33] Y. Liu, Y. Xie, and A. Srivastava, "Neural trojans," in *Proc. IEEE Int. Conf. Comput. Design (ICCD)*, Boston, MA, USA, Nov. 2017, pp. 45–48.

[34] B. G. Doan, E. Abbasnejad, and D. C. Ranasinghe, "Februus: Input purification defense against trojan attacks on deep neural network systems," in *Proc. Annu. Comput. Secur. Appl. Conf.*, Dec. 2020, pp. 897–912.

[35] E. Borgnia et al., "Strong data augmentation sanitizes poisoning and backdoor attacks without an accuracy tradeoff," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Toronto, ON, Canada, Jun. 2021, pp. 3855–3859.

[36] Y. Li et al., "Anti-backdoor learning: Training clean models on poisoned data," in *Proc. 35th Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Dec. 2021, pp. 14900–14912.

[37] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 586–595.

[38] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. 20th Int. Conf. Pattern Recognit.*, San Francisco, CA, USA, Aug. 2010, pp. 2366–2369.

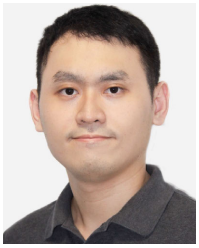
[39] W. Xue, L. Zhang, X. Mou, and A. C. Bovik, "Gradient magnitude similarity deviation: A highly efficient perceptual image quality index," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 684–695, Feb. 2014.

[40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[41] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, Jul. 2019.

[42] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014.
[43] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Lille, France, Jul. 2015, pp. 448–456.
[44] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, Lake Tahoe, NV, USA, Dec. 2012, pp. 1–9.
[45] A. F. Agarap, "Deep learning using rectified linear units (ReLU)," 2018, *arXiv:1803.08375*.
[46] A. Krizhevsky et al., "Learning multiple layers of features from tiny images," M.S. thesis, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, 2009.
[47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 248–255.
[48] P. Kaur, K. Sikka, and A. Divakaran, "Combining weakly and webly supervised learning for classifying food images," 2017, *arXiv:1712.08730*.
[49] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 3730–3738.
[50] A. Salem, R. Wen, M. Backes, S. Ma, and Y. Zhang, "Dynamic backdoor attacks against machine learning models," in *Proc. IEEE 7th Eur. Symp. Secur. Privacy (EuroSP)*, Genoa, Italy, Jun. 2022, pp. 703–718.
[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
[52] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Long Beach, CA, USA, Jun. 2019, pp. 6105–6114.
[53] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[54] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 4700–4708.
[55] L. Bottou, "Stochastic gradient descent tricks," in *Neural Networks: Tricks of the Trade*, 2nd ed. Berlin, Germany: Springer, 2012, pp. 421–436.
[56] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, pp. 303–308, Jun. 2010.

Chaohui Xu (Graduate Student Member, IEEE) received the B.Eng. degree in information engineering from Zhejiang University, Hangzhou, China, in 2020. He is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He is advised by Prof. Chip-Hong Chang on his work in trustworthy deep learning and model security.



Wenye Liu (Member, IEEE) received the B.S. degree in microelectronics from Shenzhen University, China, in 2014, the B.S. degree in physics from Umea University, Sweden, in 2014, the M.S. degree in IC design engineering from The Hong Kong University of Science and Technology, and the Ph.D. degree from Nanyang Technological University (NTU), Singapore, in 2021. He was a Research Fellow with NTU (2021–2022). His current research interests include hardware security, machine learning accelerator, and fault injection attack.

Yue Zheng (Member, IEEE) received the B.Eng. degree from the School of Communication and Information Engineering, Shanghai University (SHU), China, in 2015, and the Ph.D. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, in 2020. She was a Visiting Scholar with Kyoto University from March 2019 to June 2019. She is currently a Research Fellow with NTU. Her research interests include hardware security, physical unclonable function, security, and the privacy of artificial intelligence of things. She is a member of VLSI Systems and Applications Technical Committee (VSA-TC). She also serves as an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS.

Si Wang received the B.Eng. (Hons.) and Ph.D. degrees from the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, in 2016 and 2021, respectively. She is currently a Research Fellow with the School of Electrical and Electronic Engineering of NTU. Her research interests include hardware security and machine learning security. She has given a tutorial at VLSI Design & Embedded Systems Conference (VLSID-2022).

Chip-Hong Chang (Fellow, IEEE) received the B.Eng. degree (Hons.) from the National University of Singapore in 1989 and the M.Eng. and Ph.D. degrees from Nanyang Technological University (NTU), Singapore, in 1993 and 1998, respectively.

He is currently a Professor with the School of Electrical and Electronic Engineering (EEE), NTU. He held joint appointments with the university as the Assistant Chair of Alumni from 2008 to 2014, the Deputy Director of the Center for High Performance Embedded Systems from 2000 to 2011, and the Program Director of the Center for Integrated Circuits and Systems from 2003 to 2009. He has coedited six books, published 13 book chapters, more than 100 international journal articles (more than 80 are in IEEE) and about 200 refereed international conference papers (mostly in IEEE), and delivered over 50 keynotes, tutorials and invited seminars. His current research interests include hardware security, AI security, biometric security, trustworthy sensing, hardware accelerators for post-quantum cryptography, and edge computational intelligence. He serves as the Senior Area Editor for IEEE TRANSACTIONS ON INFORMATION FORENSIC AND SECURITY and an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS and IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS. He also served as an Associate Editor for the IEEE TRANSACTIONS ON INFORMATION FORENSIC AND SECURITY and IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS from 2016 to 2019, IEEE ACCESS from 2013 to 2019, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS from 2010 to 2013, *Integration, the VLSI Journal* from 2013 to 2015, *Journal of Hardware and System Security* (Springer) from 2016 to 2020, and *Microelectronics Journal* from 2014 to 2020. He has guest edited more than ten journal special issues and served in the organizing and technical program committee of more than 70 international conferences (mostly IEEE). He is an IET Fellow and an AAIA Fellow and a Distinguished Lecturer of IEEE Circuits and Systems Society (2018–2019).