Invisible Backdoor Attack with Sample-Specific Triggers

Yuezun Li[1], Yiming Li[4], Baoyuan Wu[2,3], Longkang Li[2,3], Ran He[5], and Siwei Lyu[6]

[1]Ocean University of China, Qingdao, China
[2]School of Data Science, The Chinese University of Hong Kong, Shenzhen, China
[3]Secure Computing Lab of Big Data, Shenzhen Research Institute of Big Data, Shenzhen, China
[4]Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, China
[5]NLPR/CRIPAC, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[6]University at Buffalo, SUNY, NY, USA

Abstract

Recently, backdoor attacks pose a new security threat to the training process of deep neural networks (DNNs). Attackers intend to inject hidden backdoors into DNNs, such that the attacked model performs well on benign samples, whereas its prediction will be maliciously changed if hidden backdoors are activated by the attacker-defined trigger. Existing backdoor attacks usually adopt the setting that triggers are sample-agnostic, i.e., different poisoned samples contain the same trigger, resulting in that the attacks could be easily mitigated by current backdoor defenses. In this work, we explore a novel attack paradigm, where backdoor triggers are sample-specific. In our attack, we only need to modify certain training samples with invisible perturbation, while not need to manipulate other training components (e.g., training loss, and model structure) as required in many existing attacks. Specifically, inspired by the recent advance in DNN-based image steganography, we generate sample-specific invisible additive noises as backdoor triggers by encoding an attacker-specified string into benign images through an encoder-decoder network. The mapping from the string to the target label will be generated when DNNs are trained on the poisoned dataset. Extensive experiments on benchmark datasets verify the effectiveness of our method in attacking models with or without defenses. The code will be available at `https://github.com/yuezunli/ISSBA`.

1. Introduction

Deep neural networks (DNNs) have been widely and successfully adopted in many areas [10, 22, 42, 17]. Large amounts of training data and increasing computational power are the key factors to their success, but the lengthy and involved training procedure becomes the bottleneck for.users and researchers. To reduce the overhead, third-party resources are usually utilized in training DNNs. For example, one can use third-party data (*e.g.*, data from the Internet or third-party companies), train their model with third-party servers (*e.g.*, Google Cloud), or even adopt third-party APIs directly. However, the opacity of the training process brings new security threats.

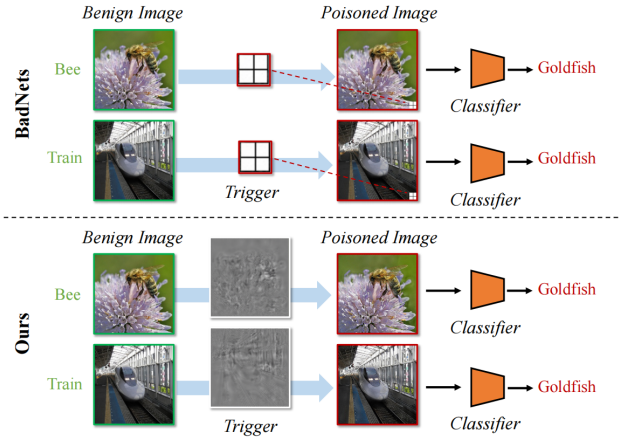


Figure 1. The comparison of triggers in previous attacks (*e.g.*, BadNets [8]) and in our attack. The triggers of previous attacks are sample-agnostic (*i.e.,* different poisoned samples contain the same trigger), while those of our method are sample-specific.

Backdoor attack1 is an emerging threat in the training process of DNNs. It maliciously manipulates the prediction of the attacked DNN model by poisoning a portion of training samples. Specifically, backdoor attackers inject some attacker-specified patterns (dubbed *backdoor triggers*) in the poisoned image and replace the corresponding label with a pre-defined *target label*. Accordingly, attackers can embed some hidden backdoors to the model trained with the poisoned training set. The attacked model will behave normally on benign samples, whereas its predic-

tion will be changed to the target label when the trigger is present. Besides, the trigger could be invisible [2, 16, 31] and the attacker only needs to poison a small fraction of samples, making the attack very stealthy. Hence, the insidious backdoor attack is a serious threat to the applications of DNNs.

Fortunately, some backdoor defenses [6, 38, 27] were proposed, which show that existing backdoor attacks can be successfully mitigated. It raises an important question: has the threat of backdoor attacks really been resolved?

In this paper, we reveal that existing backdoor attacks were easily mitigated by current defenses mostly because their backdoor triggers are *sample-agnostic*, *i.e.*, different poisoned samples contain the same trigger no matter what trigger pattern is adopted. Given the fact that the trigger is sample-agnostic, defenders can easily reconstruct or detect the backdoor trigger according to the same behaviors among different poisoned samples.

Based on this understanding, we explore a novel attack paradigm, where the backdoor trigger is *sample-specific*. We only need to modify certain training samples with invisible perturbation, while not need to manipulate other training components (*e.g.*, training loss, and model structure) as required in many existing attacks [31, 24, 25]. Specifically, inspired by DNN-based image steganography [1, 37, 35], we generate sample-specific invisible additive noises as backdoor triggers by encoding an attacker-specified string into benign images through an encoder-decoder network. The mapping from the string to the target label will be generated when DNNs are trained on the poisoned dataset. The proposed attack paradigm breaks the fundamental assumption of current defense methods, therefore can easily bypass them.

The main contributions of this paper are as follows: **(1)** We provide a comprehensive discussion about the success conditions of current main-stream backdoor defenses. We reveal that their success all relies on a prerequisite that backdoor triggers are sample-agnostic. **(2)** We explore a novel invisible attack paradigm, where the backdoor trigger is sample-specific and invisible. It can bypass existing defenses for it breaks their fundamental assumption. **(3)** Extensive experiments are conducted, which verify the effectiveness of the proposed method.

2. Related Works

2.1. Backdoor Attack

The backdoor attack is an emerging and rapidly growing research area, which poses a security threat to the training process of DNNs. Existing attacks can be categorized into two types based on the characteristics of triggers: **(1)** *visible attack* that the trigger in the attacked samples is visible for humans, and **(2)** *invisible attack* that the trigger is invisible.

Backdoor Attack . Gu *et al*. [7] first revealed the backdoor threat in the training of DNNs and proposed the BadNets attack, which is representative of visible backdoor attacks. Given an attacker-specified target label, BadNets poisoned a portion of the training images from the other classes by stamping the backdoor trigger (*e.g.*, 3×3 white square in the lower right corner of the image) onto the benign image. These poisoned images with the target label, together with other benign training samples, are fed into the DNNs for training. Currently, there was also some other work in this field [34, 19, 24]. In particular, the concurrent work [24] also studied the sample-specific backdoor attack. However, their method needs to control the training loss except for modifying training samples, which significantly reduces its threat in real-world applications.

Invisible Backdoor Attack. Chen *et al*. [2] first discussed the stealthiness of backdoor attacks from the perspective of the visibility of backdoor triggers. They suggested that poisoned images should be indistinguishable compared with their benign counter-part to evade human inspection. Specifically, they proposed an invisible attack with the blended strategy, which generated poisoned images by blending the backdoor trigger with benign images instead of by stamping directly. Besides the aforementioned methods, several other invisible attacks [28, 31, 43] were also proposed for different scenarios: Quiring *et al*. [28] targeted on the image scaling process during the training, Zhao *et al*. [43] targeted on the video recognition, and Saha *et al*. [31] assumed that attackers know model structure. Note that most of the existing attacks adopted a sample-agnostic trigger design, *i.e*., the trigger is fixed in either the training or testing phase. In this paper, we propose a more powerful invisible attack paradigm, where backdoor triggers are sample-specific.

2.2. Backdoor Defense

Pruning-based Defenses. Motivated by the observation that backdoor-related neurons are usually dormant during the inference process of benign samples, Liu *et al*. [21] proposed to prune those neurons to remove the hidden backdoor in DNNs. A similar idea was also explored by Cheng *et al*. [3], where they proposed to remove neurons with high activation values in terms of the L norm of the activation map from the final convolutional layer.

Trigger Synthesis based Defenses. Instead of eliminating hidden backdoors directly, trigger synthesis based defenses synthesize potential triggers at first, following by the second stage suppressing their effects to remove hidden backdoors. Wang *et al*. [38] proposed the first trigger synthesis based defense, *i.e.*, Neural Cleanse, where they first obtained potential trigger patterns towards every class and then determined the final synthetic trigger pattern and its target label based on an anomaly detector. Similar ideas were also stud-

ied [26, 8, 39], where they adopted different approaches for generating potential triggers or anomaly detection.

Saliency Map based Defenses. These methods used the saliency map to identify potential trigger regions to filter malicious samples. Similar to trigger synthesis based defenses, an anomaly detector was also involved. For example, SentiNet [4] adopted the Grad-CAM [32] to extract critical regions from input towards each class and then located the trigger regions based on the boundary analysis. A similar idea was also explored [12].

STRIP. Recently, Gao *et al*. [6] proposed a method, known as the STRIP, to filter malicious samples through superimposing various image patterns to the suspicious image and observe the randomness of their predictions. Based on the assumption that the backdoor trigger is input-agnostic, the smaller the randomness, the higher the probability that the suspicious image is malicious.

3. A Closer Look of Existing Defenses

In this section, we discuss the success conditions of current mainstream backdoor defenses. We argue that their success is mostly predicated on an implicit assumption that backdoor triggers are sample-agnostic. Once this assumption is violated, their effectiveness will be highly affected. The assumptions of several defense methods are discussed as follows.

The Assumption of Pruning-based Defenses . Pruningbased defenses were motivated by the assumption that backdoor-related neurons are different from those activated for benign samples. Defenders can prune neurons that are dormant for benign samples to remove hidden backdoors. However, the non-overlap between these two types of neurons holds probably because the sample-agnostic trigger patterns are simple, *i.e.*, DNNs only need few independent neurons to encode this trigger. This assumption may not hold when triggers are sample-specific, since this paradigm is more complicated.

The Assumption of Trigger Synthesis based Defenses . In the synthesis process, existing methods (*e.g.*, Neural Cleanse [38] are required to obtain potential trigger patterns that could convert any benign image to a specific class. As such, the synthesized trigger is valid only when the attack-specified backdoor trigger is sample-agnostic.

The Assumption of Saliency Map based Defenses . As mentioned in Section 2.2, saliency map based defenses required to **(1)** calculate saliency maps of all images (toward each class) and **(2)** locate trigger regions by finding universal saliency regions across different images. In the first step, whether the trigger is compact and big enough determines whether the saliency map contains trigger regions influencing the defense effectiveness. The second step requires that the trigger is sample-agnostic, otherwise, defenders can hardly justify the trigger regions.

The Assumption of STRIP . STRIP [6] examined a malicious sample by superimposing various image patterns to the suspicious image. If the predictions of generated samples are consistent, then this examined sample will be regarded as the poisoned sample. Note its success also relies on the assumption that backdoor triggers are sample-agnostic.

4. Sample-specific Backdoor Attack (SSBA)

4.1. Threat Model

Attacker's Capacities. We assume that attackers are allowed to poison some training data, whereas they have no information on or change other training components (*e.g.*, training loss, training schedule, and model structure). In the inference process, attackers can and only can query the trained model with any image. They have neither information about the model nor can they manipulate the inference process. This is the minimal requirement for backdoor attackers [18]. The discussed threat can happen in many real-world scenarios, including but not limited to adopting third-party training data, training platforms, and model APIs.

Attacker's Goals . In general, backdoor attackers intend to embed hidden backdoors in DNNs through data poisoning. The hidden backdoor will be activated by the attacker-specified trigger, *i.e.*, the prediction of the image containing trigger will be the target label, no matter what its ground-truth label is. In particular, attackers has three main goals, including the *effectiveness*, *stealthiness*, and *sustainability*. The *effectiveness* requires that the prediction of attacked DNNs should be the target label when the backdoor trigger appears, and the performance on benign testing samples will not be significantly reduced; The *stealthiness* requires that adopted triggers should be concealed and the proportion of poison samples (*i.e.*, the poisoning rate) should be small; The *sustainability* requires that the attack should still be effective under some common backdoor defenses.

4.2. The Proposed Attack

In this section, we illustrate our proposed method. Before we describe how to generate sample-specific triggers, we first briefly review the main process of attacks and present the definition of a sample-specific backdoor attack.

The Main Process of Backdoor Attacks . Let $D_{train} = \{(x_i, y_i)\}_{i=1}^{N}$ indicates the benign training set containing N *i.i.d.* samples, where $x_i \in \mathcal{X} = \{0, , 255\}^{C\times W\times H}$ and $y_i \in \mathcal{Y} = \{1, , K\}$. The classification learns a function $f_w : \mathcal{X} = [0, 1]^K$ with parameters $\boldsymbol{w}$. Let y_t denotes the target label ($y_t \in \mathcal{Y}$). The core of backdoor attacks is how to generate the *poisoned training set* D_p. Specifically, D_p consists of modified version of a subset of D_{train} (*i.e.*, D_m) and remaining benign samples D_b, *i.e.*,

$$D_p = D_m \cup D_b \tag{1}$$

where $D_b \subset D_{train}$, $\gamma = \frac{|D_m|}{|D_{train}|}$ indicates the poisoning rate, $D_m = \left\{(x^{'}, y_t)|x^{'} = G_\theta(x), (x, y) \in D_{train} \setminus D_b\right\}$, $G_\theta : \mathcal{X} \to \mathcal{X}$ is an attacker-specified poisoned image generator. The smaller the γ, the more stealthy the attack.

Definition 1 . *A backdoor attack with poisoned image generator* $G()$ is called sample-specific if and only if $\forall x_i, x_j \in \mathcal{X}(x_i \neq x_j), T(G(x_i)) \neq T(G(x_j))$, where $T(G(x))$ indicates the backdoor trigger contained in the *poisoned sample* $G(\boldsymbol{x})$.

Remark 1 . *Triggers of previous attacks are not sample-specific. For example, for the attack proposed in [3],* $T(G(x)) = t, \forall x \in \mathcal{X}$, *where* $G(x) = (1 - \lambda) \bigotimes x + \lambda \bigotimes t$.

How to Generate Sample-specific Triggers . {We use a pre-trained encoder-decoder network as an example to generate sample-specific triggers, motivated by the DNN-based image steganography [1, 37, 35]. The generated triggers are invisible additive noises containing a representative string of the target label. The string can be flexibly designed by the attacker. For example, it can be the name, the index of the target label, or even a random character. As shown in Figure 2, the encoder takes a benign image and the representative string to generate the poisoned image (*i.e.*, the benign image with their corresponding trigger). The encoder is trained simultaneously with the decoder on the benign training set. Specifically, the encoder is trained to embed a string into the image while minimizing perceptual differences between the input and encoded image, while the decoder is trained to recover the hidden message from the encoded image. Their training process is demonstrated in Figure 3. Note that attackers can also use other methods, such as VAE[23] [17], to conduct the sample-specific backdoor attack. It will be further studied in our future work.

Pipeline of Sample-specific Backdoor Attack . Once the poisoned training set $D_{poisoned}$ is generated based on the aforementioned method, backdoor attackers will send it to the user. Users will adopt it to train DNNs with the standard training process, *i.e.,*

$$\min_{w} \frac{1}{N} \sum_{(x,y) \in D_{poisoned}} \mathcal{L}(f_w(x), y) \tag{2}$$

where $\mathcal{L}$ indicated the loss function, such as the cross-entropy. The optimization (2) can be solved by back-propagation [30] with the stochastic gradient descent [41]. The mapping from the representative string to the target label will be learned by DNNs during the training process. Attackers can activate hidden backdoors by adding triggers to the image based on the encoder in the inference stage.

5. Experiments

5.1. Experimental Settings

Datasets and Models. We consider two classical image classification tasks: **(1)** object classification, and **(2)** face recognition. For the first task, we conduct experiments on the ImageNet [5] dataset. For simplicity, we randomly select a subset containing 200 classes with 100, 000 images for training (500 images per class) and 10, 000 images for testing (50 images per class). The image size is 3 × 224 × 224. Besides, we adopt MS-Celeb-1M dataset [9] for face recognition. In the original dataset, there are nearly 100,000 identities containing different numbers of images ranging from 2 to 602. For simplicity, we select the top 100 identities with the largest number of images. More specifically, we obtain 100 identities with 38,000 images (380 images per identity) in total. The split ratio of training and testing sets is set to 8:2. For all the images, we firstly perform face alignments, then select central faces, and finally resize them into 3×224×224. We use ResNet-18 [10] as the model structure for both datasets. More experiments with VGG-16 [33] are in the supplementary materials.

Baseline Selection. We compare the proposed sample-specific backdoor attack with BadNets [7] and the typical invisible attack with blended strategy (dubbed *Blended Attack*) [2]. We also provide the model trained on the benign dataset (dubbed *Standard Training*) as another baseline for reference. Besides, we select Fine-Pruning [21], Neural Cleanse [38], SentiNet [4], STRIP [6], DF-TND [39] , and Spectral Signatures [36] to evaluate the resistance to state-of-the-art defenses.

Attack Setup. We set the poisoning rate = 10% and target label yt = 0 for all attacks on both datasets. As shown in Figure 4, the backdoor trigger is a 20×20 white-square with a cross-line on the bottom right corner of poisoned images for both BadNets and Blended Attack, and the trigger transparency is set to 10% for the Blended Attack. The triggers of our methods are generated by the encoder trained on the benign training set. Specifically, we follow the settings of the encoder-decoder network in StegaStamp [35], where we use a U-Net [29] style DNN as the encoder, a spatial transformer network [14] as the decoder, and four loss-terms for the training: $L2$ residual regularization, LPIPS perceptual loss [40], a critic loss, to minimize perceptual distortion on encoded images, and a cross-entropy loss for code reconstruction. The scaling factors of four loss-terms are set to 2.0, 1.5, 0.5, and 1.5. For the training of all encoder-decoder networks, we utilize Adam optimizer [15] and set the initial learning rate as 0.0001. The batch size and training iterations are set to 16 and 140, 000, respectively. Moreover, in the training stage, we utilize the SGD optimizer and set the initial learning rate as 0.001. The batch size and maximum epoch are set as 128 and 30, respectively. The learning rate

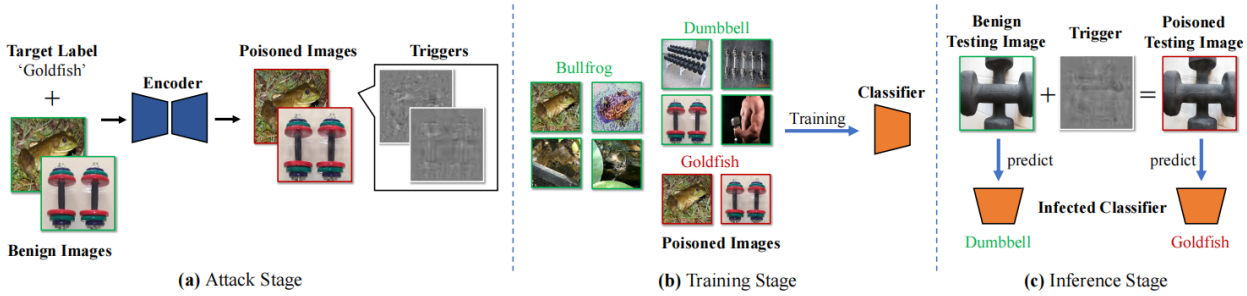


Figure 2. The pipeline of our attack. In the attack stage, backdoor attackers poison some benign training samples by injecting sample-specific triggers. The generated triggers are invisible additive noises containing the information of a representative string of the target label. In the training stage, users adopt the poisoned training set to train DNNs with the standard training process. Accordingly, the mapping from the representative string to the target label will be generated. In the inference stage, infected classifiers (*i.e.*, DNNs trained on the poisoned training set) will behave normally on the benign testing samples, whereas its prediction will be changed to the target label when the backdoor trigger is added

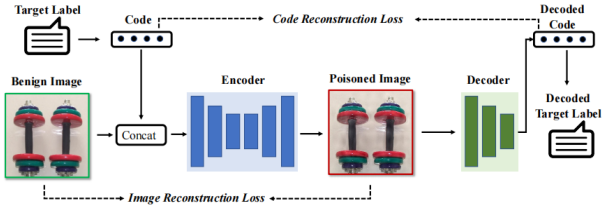


Figure 3. The training process of encoder-decoder network. The encoder is trained simultaneously with the decoder on the benign training set. Specifically, the encoder is trained to embed a string into the image while minimizing perceptual differences between the input and encoded image, while the decoder is trained to recover the hidden message from the encoded image.

is decayed with factor 0.1 after epoch 15 and 20.

Defense Setup.. For Fine-Pruning, we prune the last convolutional layer of ResNet-18 (Layer4.conv2); For Neural Cleanse, we adopt its default setting and utilize the generated anomaly index for demonstration. The smaller the value of the anomaly index, the harder the attack to defend; For STRIP, we also adopt its default setting and present the generated entropy score. The larger the score, the harder the attack to defend; For SentiNet, we compared the generated Grad-CAM [32] of poisoned samples for demonstration; For DF-TND, we report the logit increase scores before and after the universal adversarial attack of each class. This defense succeeds if the score of the target label is significantly larger than those of all other classes. For Spectral Signatures, we report the outlier score for each sample, where a larger score denotes the sample is more likely poisoned.

Evaluation Metric. We use the attack success rate (ASR) and benign accuracy (BA) to evaluate the effectiveness of different attacks. Specifically, ASR is defined as the ratio between successfully attacked poison samples and total poison samples. BA is defined as the accuracy of testing on benign samples. Besides, we adopt the peak-signal-to noise-ratio (PSNR) [13] and L_∞ norm [11] to evaluate the stealthiness.

5.2. Main Results

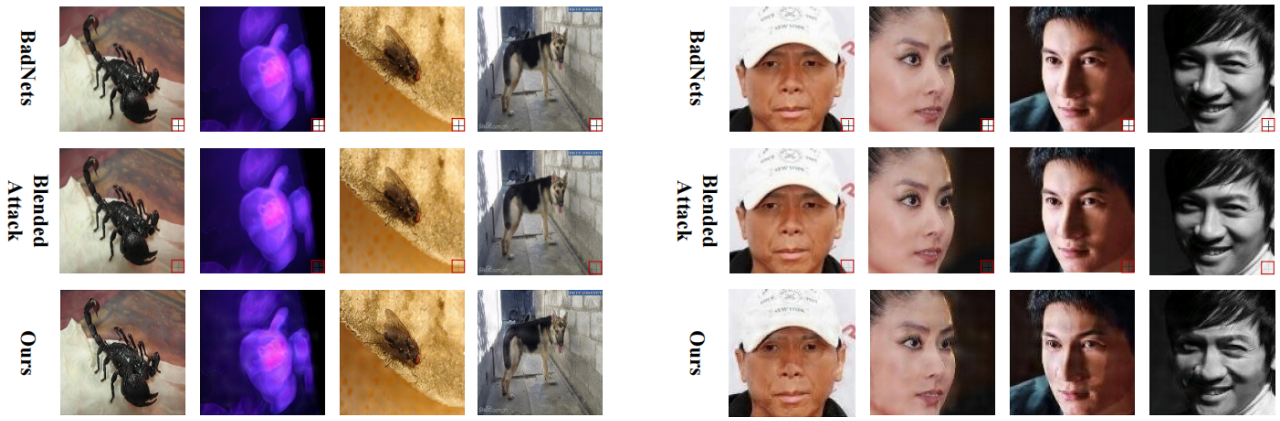
Attack Effectiveness. As shown in Table 1, our attack can successfully create backdoors with a high ASR by poisoning only a small proportion (10%) of training samples. Specifically, our attack can achieve an ASR ¿ 99% on both datasets. Besides, the ASR of our method is on par with that of BadNets and higher than that of Blended Attack. Moreover, the accuracy reduction of our attack (compared with the Standard Training) on benign testing samples is less than 1% on both datasets, which are smaller than those of BadNets and Blended Attack. These results show that sample-specific invisible additive noises can also serve as good triggers even though they are more complicated than the white-square used in BadNets and Blended Attack.

Attack Stealthiness. Figure 4 presents some poisoned images generated by different attacks. Although our attack does not achieve the best stealthiness regarding PSNR and L_∞ we are the second-best, as shown in Table 1), poisoned images generated by our method still look natural to the human inspection. Although Blended Attack seems to have the best stealthiness regarding adopted evaluation metrics, triggers in their generated samples still quite obvious, especially when the background is dark.

Resistance to Fine-Pruning. In this part, we compare our attack to BadNets and Blended Attack in terms of the resistance to the pruning-based defense [21]. As shown in Figure 5, the ASR of BadNets and Blended Attack drop dramatically when only 20% of neurons are pruned. Especially the Blended Attack, its ASR decrease to less than 10% on both

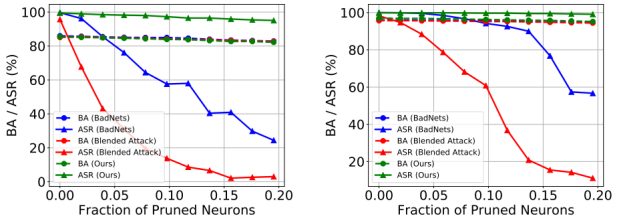
Table 1. The comparison of different methods against DNNs without defense on the ImageNet and MS-Celeb-1M dataset. Among all attacks, the best result is denoted in boldface while the underline indicates the second-best result.

Dataset →	ImageNet				MS-Celeb-1M			
Aspect →	Effectiveness (%)		Stealthiness		Effectiveness (%)		Stealthiness	
Attack ↓	BA	ASR	PSNR	L	BA	ASR	PSNR	L
Standard Training	85.8	0.0	-	-	97.3	0.1	-	-
BadNets [7]	85.9	99.7	25.635	235.583	96.0	100	25.562	229.675
Blended Attack [2]	85.1	95.8	45.809	23.392	95.7	99.1	45.726	23.442
Ours	85.5	99.5	27.195	83.198	96.5	100	28.659	91.071



(a) ImageNet (b) MS-Celeb-1M

Figure 4. Poisoned samples generated by different attacks. BadNets and Blended Attack use a white-square with the cross-line (**areas in the red box**) as the trigger pattern, while triggers of our attack are sample-specific invisible additive noises on the whole image.



(a) ImageNet (b) MS-Celeb-1M

Figure 5. Benign accuracy (BA) and attack success rate (ASR) of different attacks against pruning-based defense.

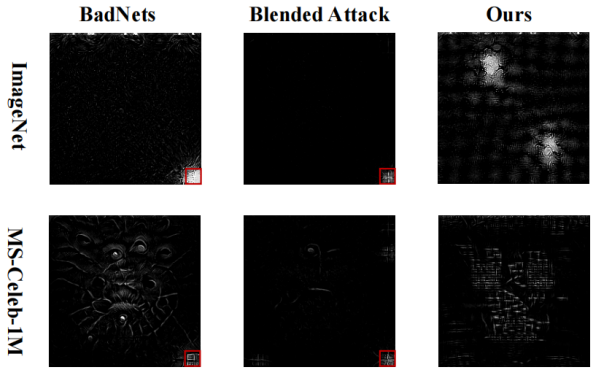
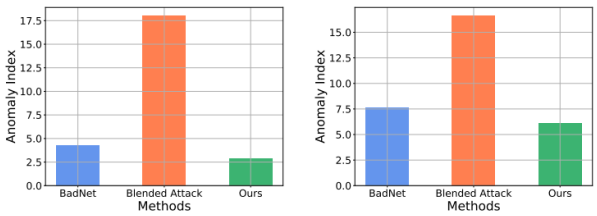


Figure 6. The synthesized triggers generated by Neural Cleanse. The red box in the figure indicates ground-truth trigger areas.

ImageNet and MS-Celeb-1M datasets. In contrast, the ASR of our attack only decreases slightly (less than 5%) with the increase of the fraction of pruned neurons. Our attack retains an ASR greater than 95% on both datasets when 20% of neurons are pruned. This suggests that our attack is more resistant to the pruning-based defense.

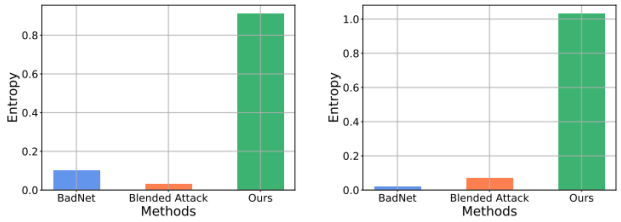
Resistance to Neural Cleanse. Neural Cleanse [38] computes the trigger candidates to convert all benign images to each label. It then adopts an anomaly detector to verify whether anyone is significantly smaller than the others as the backdoor indicator. The smaller the value of the anomaly index, the harder the attack for Neural-Cleanse to defend. As shown in Figure 7, our attack is more resistant to the Neural-Cleanse. Besides, we also visualize the synthesized trigger (*i.e.*, the one with the smallest anomaly index among all candidates) of different attacks. As shown in Figure 6, synthesized triggers of BadNets and Blended Attack contain similar patterns to those used by attackers (*i.e.*, white-square on the bottom right corner), whereas those of our attack are meaningless.

Resistance to STRIP. STRIP [6] filters poisoned samples based on the prediction randomness of samples generated by imposing various image patterns on the suspicious image. The randomness is measured by the entropy of the



(a) ImageNet (b) MS-Celeb-1M

Figure 7. The anomaly index of different attacks. The smaller the index, the harder the attack for Neural-Cleanse to defend.



(a) ImageNet (b) MS-Celeb-1M

Figure 8. The entropy generated by STRIP of different attacks. The higher the entropy, the harder the attack for STRIP to defend.

Table 2. The ASR (%) of our attack with consistent (dubbed *Ours*) or inconsistent (dubbed *Ours (inconsistent)*) triggers. The inconsistent trigger is generated based on a different testing image.

	ImageNet	MS-Celeb-1M
ours	99.5	100
Ours (inconsistent)	23.3	98.1

average prediction of those samples. As such, the higher the entropy, the harder an attack for STRIP to defend. As shown in Figure 8, our attack is more resistant to the STRIP compared with other attacks.

Resistance to SentiNet.. SentiNet [4] identities trigger regions based on the similarities of Grad-CAM of different samples. As shown in Figure 9, Grad-CAM successfully distinguishes trigger regions of those generated by BadNets and Blended Attack, while it fails to detect trigger regions of those generated by our attack. In other words, our attack is more resistant to SentiNet.

5.3. Discussion

In this section, unless otherwise specified, all settings are the same as those stated in Section 5.1.

The Exclusiveness of Generated Triggers. In this part, we explore whether the generated sample-specific triggers are exclusive, *i.e.,* whether testing image with trigger generated based on another image can also activate the hidden backdoor of DNNs attacked by our method. Specifically, for each testing image x, we randomly select another testing image $x^{'}(x^{'} \neq x)$. Now we query the attacked DNNs with $x + T(G(x^{'}))$ (rather than with $x + T(G(x))$). As shown in Table 2, the ASR decreases sharply when inconsistent triggers (*i.e.*, triggers generated based on different images) are adapted on the ImageNet dataset. However, on the MS-Celeb-1M dataset, attacking with inconsistent triggers can still achieve a high ASR. This may probably be because most of the facial features are similar and therefore the learned trigger has better generalization. We will further explore this interesting phenomenon in our future work.

Table 3. Out-of-dataset generalization of our method in the attack stage. See text for details.

Dataset for Classifier →	ImageNet		MS-Celeb-1M	
Dataset for Encoder ↓	BA	ASR	BA	ASR
ImageNet	85.5	99.5	95.6	99.5
MS-Celeb-1M	85.1	99.4	96.5	100

Table 4. The ASR (%) of our method attacked with out-of-dataset testing samples. See text for details.

Dataset for Training → Dataset for Inference ↓	ImageNet	MS-Celeb-1M
Microsoft COCO	100	99.9
Random Noise	100	99.9

Out-of-dataset Generalization in the Attack Stage. Recall that the encoder is trained on the benign version of the poisoned training set in previous experiments. In this part, we explore whether the one trained on another dataset can still be adapted for generating poisoned samples of a new dataset (without any fine-tuning) in our attack. As shown in Table 3, the effectiveness of attack with an encoder trained on another dataset is on par with that of the one trained on the same dataset. In other words, attackers can reuse already trained encoders to generate poisoned samples, if their image size is the same. *This property will significantly reduce the computational cost of our attack.*

Out-of-dataset Generalization in the Inference Stage. In this part, we verify that whether out-of-dataset images (with triggers) can successfully attack DNNs attacked by our method. We select the Microsoft COCO dataset [20] and a synthetic noise dataset for the experiment. They are representative of nature images and synthetic images, respectively. Specifically, we randomly select 1,000 images from the Microsoft COCO and generate 1,000 synthetic images where each pixel value is uniformly and randomly selected from $\{0, , 255\}$. All selected images are resized to $3\times224\times224$. As shown in Table 4, our attack with poisoned samples generated based on out-of-dataset images can also achieve nearly 100% ASR. *It indicates that attackers can activate the hidden backdoor in attacked DNNs with out-of-dataset images (not necessary with testing images).*

6. Conclusions

In this paper, we showed that existing backdoor attacks were easily alleviated by current backdoor defenses mostly

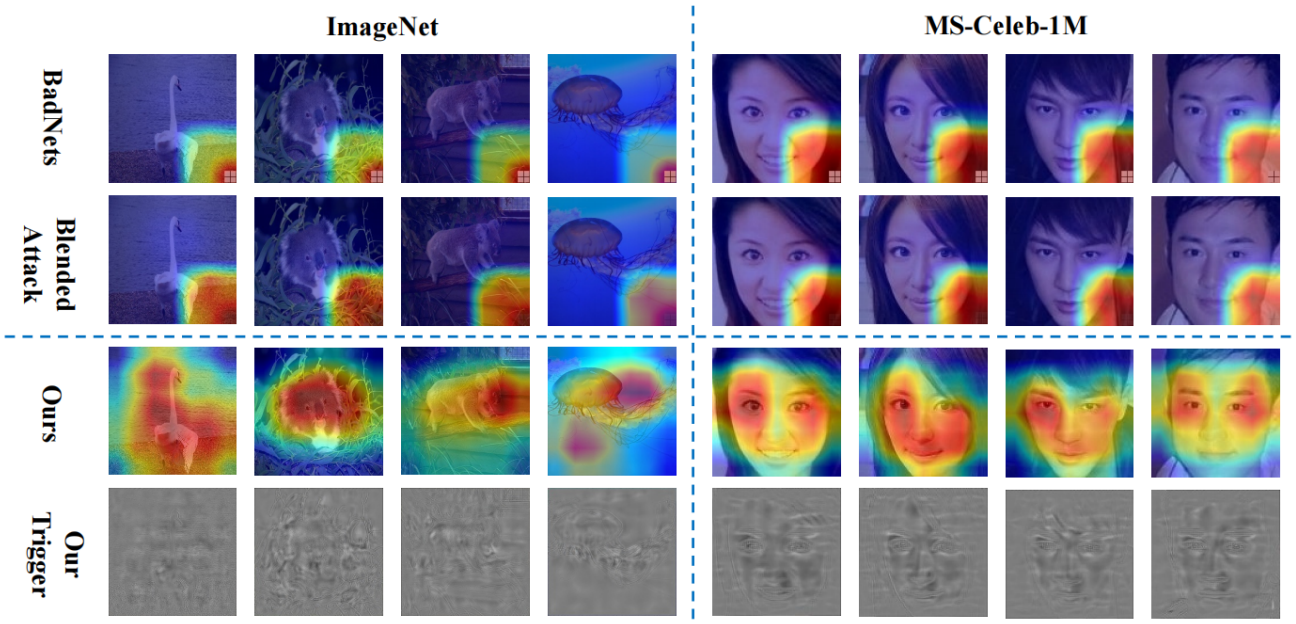


Figure 9. The Grad-CAM of poisoned samples generated by different attacks. As shown in the figure, Grad-CAM successfully distinguishes trigger regions of those generated by BadNets and Blended Attack, while it fails to detect trigger regions of those generated by our attack.

because their backdoor trigger is sample-agnostic, *i.e.*, different poisoned samples contain the same trigger. Based on this understanding, we explored a new attack paradigm, the sample-specific backdoor attack (SSBA), where the backdoor trigger is sample-specific. Our attack broke the fundamental assumption of defenses, therefore can bypass them. Specifically, we generated sample-specific invisible additive noises as backdoor triggers by encoding an attacker-specified string into benign images, motivated by the DNN-based image steganography. The mapping from the string to the target label will be learned when DNNs are trained on the poisoned dataset. Extensive experiments were conducted, which verify the effectiveness of our method in attacking models with or without defenses.

7. Acknowledgement

Yuezun Li is supported in part by China Postdoc Science Foundation under grant No.2021TQ0314. Baoyuan Wu is supported by the Natural Science Foundation of China under grant No.62076213, the university development fund of the Chinese University of Hong Kong, Shenzhen under grant No.01001810, the special project fund of Shenzhen Research Institute of Big Data under grant No.T00120210003, and Shenzhen Science and Technology Program under grant No.GXWD20201231105722002-20200901175001001. Siwei Lyu is supported by the Natural Science Foundation under grants No.IIS-2103450 and IIS-1816227.

References

[1] Shumeet Baluja. Hiding images in plain sight: Deep steganography. 2017. 2, 4

[2] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. 2017. 2, 4, 6

[3] Hao Cheng, Kaidi Xu, Sijia Liu, Pin Yu Chen, and Xue Lin. Defending against backdoor attack on deep neural networks. 2020. 2

[4] Edward Chou, Florian Tramèr, and Giancarlo Pellegrino. Sentinet: Detecting localized universal attacks against deep learning systems. 2018. 3, 4, 7

[5] Jia Deng, Wei Dong, R. Socher, Li Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *Proc of IEEE Computer Vision Pattern Recognition*, pages 248–255, 2009. 4

[6] Yansong Gao, Chang Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. 2019. 2, 3, 4, 6

[7] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 2019. 2, 4, 6

[8] Wenbo Guo, Lun Wang, Yan Xu, Xinyu Xing, and Dawn Song. Towards inspecting and eliminating trojan backdoors in deep neural networks. In *2020 IEEE International Conference on Data Mining (ICDM)*, 2020. 3

[9] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. *European Conference on Computer Vision*, 2016. 4

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *IEEE*, 2016. 1, 4

[11] R. Hogg, J. Mckean, and A. T. Craig. Introduction to mathematical statistics (6th edition). 2005. 5

[12] Xijie Huang, Moustafa Alzantot, and Mani Srivastava. Neuroninspect: Detecting backdoors in neural networks via output explanations. 2019. 3

[13] Q. Huynh-Thu and M. Ghanbari. Scope of validity of psnr in image/video quality assessment. *Electronics Letters*, 44(13):800–801, 2008. 5

[14] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep structured output learning for unconstrained text recognition. In *International Conference on Learning Representations*, 2014. 4

[15] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Computer Science*, 2014. 4

[16] Shaofeng Li, Minhui Xue, Benjamin Zi Hao Zhao, Haojin Zhu, and Xinpeng Zhang. Invisible backdoor attacks on deep neural networks via steganography and regularization. *IEEE*, (5), 2021. 2

[17] Xin Li, Chao Ma, Baoyuan Wu, Zhenyu He, and Ming Hsuan Yang. Target-aware deep tracking. *IEEE*, 2019. 1

[18] Yiming Li, Baoyuan Wu, Yong Jiang, Zhifeng Li, and Shu Tao Xia. Backdoor learning: A survey. 2020. 3

[19] Junyu Lin, Lei Xu, Yingqi Liu, and Xiangyu Zhang. Composite backdoor attack for deep neural network by mixing existing benign features. In *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020. 2

[20] Tsung Yi Lin, Michael Maire, Serge Belongie, James Hays, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. *Springer International Publishing*, 2014. 7

[21] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Research in Attacks, Intrusions, and Defenses*, 2018. 2, 4, 5

[22] Li Liu, Gang Feng, Denis Beautemps, and Xiao Ping Zhang. Re-synchronization using the hand preceding model for multi-modal fusion in automatic continuous cued speech recognition. *Institute of Electrical and Electronics Engineers (IEEE)*, 2021. 1

[23] Romain Lopez, Pierre Boyeau, Nir Yosef, Michael I. Jordan, and Jeffrey Regier. Decision-making with auto-encoding variational bayes. 2020. 4

[24] Anh Nguyen and Anh Tran. Input-aware dynamic backdoor attack, 2020. 2

[25] Anh Nguyen and Anh Tran. Wanet – imperceptible warping-based backdoor attack. 2021. 2

[26] Ximing Qiao, Yukun Yang, and Hai Li. Defending neural backdoors via generative distribution modeling. 2019. 3

[27] Han Qiu, Yi Zeng, Shangwei Guo, Tianwei Zhang, Meikang Qiu, and Bhavani Thuraisingham. Deepsweep: An evaluation framework for mitigating dnn backdoor attacks using data augmentation. 2020. 2

[28] Erwin Quiring and Konrad Rieck. Backdooring and poisoning neural networks with image-scaling attacks. 2020. 2

[29] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015. 4

[30] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. *Readings in Cognitive Science*, 323(6088):399–421, 1988. 4

[31] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. pages 11957–11965, 2020. 2

[32] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *IEEE International Conference on Computer Vision*, 2017. 3, 5

[33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv e-prints*, 2014. 4

[34] Te Juin Lester Tan and Reza Shokri. Bypassing backdoor detection algorithms in deep learning. In *2020 IEEE European Symposium on Security and Privacy (EuroSP)*, 2020. 2

[35] Matthew Tancik, Ben Mildenhall, and Ren Ng. Stegastamp: Invisible hyperlinks in physical photographs. In *Computer Vision and Pattern Recognition*, 2020. 2, 4

[36] Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. 2018. 4

[37] Bruce H. Turner. Nail plate and cable protection sleeve for building framing. 2, 4

[38] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y. Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks*. 2, 3, 4, 6

[39] Ren Wang, Gaoyuan Zhang, Sijia Liu, Pin Yu Chen, Jinjun Xiong, and Meng Wang. Practical detection of trojan neural networks: Data-limited and data-free cases. 2020. 3, 4

[40] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *IEEE*, 2018. 4

[41] Tong Zhang. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, 2004. 4

[42] Yuyang Zhang, Shibiao Xu, Baoyuan Wu, Jian Shi, and Xiaopeng Zhang. Unsupervised multi-view constrained convolutional network for accurate depth estimation. *IEEE Transactions on Image Processing*, PP(99):1–1, 2020. 1

[43] Shihao Zhao, Xingjun Ma, Xiang Zheng, James Bailey, and Yu Gang Jiang. Clean-label backdoor attacks on video recognition models. 2020. 2