3S-Attack: Spatial, Spectral and Semantic Invisible Backdoor Attack Against DNN Models

Jianyao Yin[1]*, Luca Arnaboldi[1], Honglong Chen[2], Pascal Berrang[1]
[1]University of Birmingham, Birmingham, UK
[2]China University of Petroleum (East China), Qingdao, Shandong, China

Abstract

Backdoor attacks involve either poisoning the training data or directly modifying the model in order to implant a hidden behavior, that causes the model to misclassify inputs when a specific trigger is present. During inference, the model maintains high accuracy on benign samples but misclassifies poisoned samples into an attacker-specified target class. Existing research on backdoor attacks has explored developing triggers in the spatial, spectral (frequency), and semantic (feature) domains, aiming to make them stealthy. While some approaches have considered designing triggers that are imperceptible in both spatial and spectral domains, few have incorporated the semantic domain. In this paper, we propose a novel backdoor attack, termed 3S-attack, which is stealthy across the spatial, spectral, and semantic domains. The key idea is to exploit the semantic features of benign samples as triggers, using Gradient-weighted Class Activation Mapping (Grad-CAM) and a preliminary model for extraction. The trigger is then embedded in the spectral domain, followed by pixel-level restrictions after converting the samples back to the spatial domain. This process minimizes the distance between poisoned and benign samples, making the attack harder to detect by existing defenses and human inspection. Extensive experiments on various datasets, along with theoretical analysis, demonstrate the stealthiness of 3S-attack and highlight the need for stronger defenses to ensure AI security. Our code is available at: `https://anonymous.4open.science/r/anon-project-3776/`

Keywords: Artificial intelligence Security, Backdoor attack, Deep neural network, DCT transform

1 Introduction

With the rapid integration of artificial intelligence (AI) into diverse sectors such as finance, healthcare, and daily life, concerns about the security and trustworthiness of AI systems are intensifying. An increasing number of studies have revealed the vulnerabilities of AI models, raising concerns about their reliability in real-world applications. Based on the attackers' objectives, the attacks targeting AI model can be broadly classified into model manipulation attacks [1] and data extraction attacks [2]. Model manipulation attacks aim to disrupt the inference process of AI models, forcing them to produce unexpected outputs or attacker-specified results. For example, adversarial attacks [3] and scaling attacks [4] modify benign samples in subtle ways, causing the model to misclassify them. Backdoor attacks [5], on the other hand, implant hidden triggers in the model through poisoned training data or direct modifications to the model, enabling the attacker to manipulate the model's output whenever these triggers are present. Similarly, data poisoning attacks [6] disrupt the learning process by altering the labels of certain training samples, degrading the model's ability to generalize accurately. In contrast, data extraction attacks [2] focus on stealing information about the model or its training data. By analyzing the inputs and outputs of a model or its internal parameters, attackers can extract sensitive information. These attacks can reveal whether a specific sample was part of the model's training data (membership inference) [7], reconstruct representative samples for each class (model inversion) [8], or infer detailed properties of the training data (attribute inference) [9]. In some cases, attackers can even recover specific training samples (data reconstruction) [10].

Among them, backdoor attacks have drawn significant attention due to their stealthy nature and minimal deployment cost [11]. In a typical backdoor attack, an adversary poisons a small subset of the training data by injecting inputs containing a specific trigger and labeling them with the target class. Once trained, the model performs well on benign inputs but misclassifies any input with the trigger into the attacker-specified class. Notably, modifying as little as 1% of the training data is sufficient to embed a backdoor [11], and the entanglement of backdoor functionality with normal neurons further complicates detection and removal. Consequently, defending against such attacks remains a pressing challenge.

Over the years, various defense strategies have emerged, targeting different domains: spatial [12], spectral [13], and semantic [14] characteristics. In neural networks, spatial domain refers to the arrangement of pixels in an image, spectral domain focuses on frequency components of samples (e.g., via Fourier transforms), and semantic domain captures latent features of sample generated by pre-defined metrics or the model. In response, attackers have developed more covert strategies, seeking to evade these defenses by optimizing the stealthiness of the trigger across specific domains [15, 16]. However, stealthiness in the semantic (feature) domain—which closely reflects the model's internal representation—has received limited attention. Existing semantic-aware attacks either require access to the training process [17], or fail to achieve stealth across multiple domains simultaneously.

To address these limitations, we propose 3S-Attack, a novel backdoor attack that achieves stealthiness across three complementary domains: spatial, spectral, and semantic. Our method does not require access to the training pipeline. Instead, it operates solely through data poisoning. Leveraging Grad-CAM [18], we extract the semantic features of benign class samples and embed them into the poisoned images. We then restrict pixel-level perturbations to preserve visual indistinguishability. The resulting poisoned samples remain nearly identical to clean ones in appearance, frequency characteristics, and high-level features, effectively evading both human perception and

*Corresponding author: Jianyao.yin@outlook.com

state-of-the-art defense techniques.

As illustrated in Figure 1, 3S-Attack introduces less perturbation to both spatial space and spectral space compared to widely adopted backdoor methods, while achieving strong attack success rates.

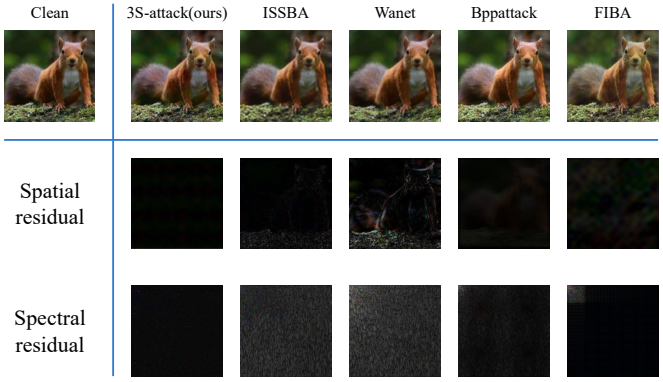

Figure 1: Comparison of proposed 3S-attack with other state-of-the-art (SOTA) backdoor attacks in spatial and spectral perspective.

The main contributions of this work are as follows:

1. We systematically analyze the limitations of existing backdoor attacks and defenses across different domains.
2. We propose 3S-Attack, the first backdoor attack to simultaneously achieve stealthiness in spatial, spectral, and semantic domains.
3. 3S-Attack is also the first semantic-domain stealthy backdoor attack that operates purely through poisoned samples, without requiring access to the model training process.
4. Extensive experiments and theoretical analysis demonstrate the superior stealth and defense-resistance capabilities of our proposed method compared to prior state-of-the-art attacks.

The rest of this paper is structured as follows. Section 2 reviews related work on backdoor attacks and defenses. Section 3 introduces the attack design and methodology. Section 4 presents experimental evaluations and analysis. Finally, Section 6 concludes the paper.

2 Background and Related Work

Research on backdoor attacks can be divided into two categories: attack schemes and corresponding defense methods.

2.1 Existing Backdoor Attacks

In 2017, Gu et al. [11] first proposed BadNets, defining the concept of backdoor attacks targeting DNN models and revealing their potential risks. In BadNets, the attacker first determines a trigger pattern and its location, as well as a corresponding target class. Then, a batch of samples from non-target classes in the training dataset is randomly selected, embedded with the trigger, and their labels are modified to the target class. These generated samples are referred to as poisoned samples, which are subsequently injected back into the training dataset. When a user trains a model using this dataset, the backdoor linking the trigger to the target class is implanted into the model. Since then, a plethora of research papers on backdoor attacks have emerged. Currently, research on backdoor attacks can be categorized into two stages: visible backdoor attacks and invisible backdoor attacks.

Visible Attacks At this stage, attackers primarily focus on enhancing the reliability and attack success rate (ASR) of the attack, paying less attention to whether the trigger is conspicuous, i.e., whether it can be detected by human observation or defense methods. Chen et al. [19] proposed two types of backdoor attacks: a single-instance-key attack, which aims to mislead the model into recognizing any sample of one person as another; and a pattern-key attack, which causes the model to misclassify any sample containing a specific pattern. Barni et al. [20] introduced a clean-label backdoor attack in which the attacker achieves the backdoor effect by modifying only the samples without changing their labels. Lovisotto et al. [21] extended backdoor attacks to biometric systems, providing practical application scenarios and highlighting that, if left undefended, backdoor attacks could be deployed in nearly all deep learning settings. Liu et al. [22] proposed a black-box backdoor attack aimed at reducing the usability of DNN models.

Invisible Attacks After the concept of backdoor attacks was introduced, numerous researchers began developing defense methods. Consequently, as understanding of backdoor attacks deepened, human-recognizable triggers were gradually abandoned due to their susceptibility to detection.

During this stage, researchers not only ensured the effectiveness of the attack but also emphasized improving its stealthiness. This includes invisibility to defense methods, i.e., bypassing various defenses, and invisibility to humans, ensuring the poisoned samples appear normal and coherent to the human eye. Specifically, Xue et al. [23] proposed two types of backdoor attacks: one-to-N attacks, where one type of trigger activates multiple backdoors, and N-to-one attacks, where multiple triggers activate one backdoor, both of which bypass various defense methods. Liao et al. [24] introduced the first global trigger backdoor attack by overlaying a shallow watermark as the trigger, achieving high attack success rates while remaining imperceptible to humans. Zou et al. [25] proposed inserting one or more neural-level trojans into the neural network, achieving high stability, as only images with triggers could activate the trojan (or backdoor). Wang et al. [26] proposed an invisible trigger based on existing biological studies of the human visual system and the assumption that the human eye is insensitive to small color differences. This attack generates triggers by re-encoding and dithering samples, making them undetectable by humans and defense methods.

Subsequently, researchers found that adding backdoors to the frequency domain features of samples can make the poisoned samples inherently covert in the spatial domain. Therefore, some researchers have attempted to implement backdoor attacks from the frequency perspective. For example, Yu et al. [27] proposed applying a Fast Fourier Transform (FFT) to the samples, adding the features of trigger samples to target samples, and then performing an inverse transform to obtain poisoned samples. Gao et al. [28] proposed a dual stealthy backdoor attack (DUBA) that embeds the trigger via discrete wavelet transform (DWT), then smooths the trigger by applying a Fast Fourier Transform (FFT) and Discrete Cosine Transform (DCT), making the trigger stealthy in both the spatial and spectral domains. Zeng et al. [13] proposed a trigger pattern that smoothly changes the pixel values in the spatial domain, which avoids abnormal spikes in the frequency map.

Other Backdoor Attacks In addition to adding poisoned samples to training datasets, researchers have also explored implanting backdoors by directly modifying models. Tang et al. [29] achieved the association between triggers and target classes by adding a bypass from input to output in a trained model, enabling backdoor implantation without modifying the dataset. Doan et al. [30] proposed that during the training phase, when model parameter information is stored in memory, accessing and modifying specific neuron parameters via programs can also implant backdoors into the model. If the attacker is a provider of Machine Learning as a Service (MLaaS), they can access both the training dataset and the training process, enabling more efficient and stealthy backdoor attacks. Liu et al. [31] suggested first performing reverse engineering on the model to obtain an initial trigger, then fine-tuning the model to enhance the trigger's attack success rate. Zhong et al. [17] proposed using a U-net encoder to generate poisoned samples. The attacker modifies the model's loss function and trains the U-net synchronously with the model to generate optimal triggers, significantly improving the attack's stealthiness.

Beyond image classification tasks, recent work has extended backdoor attacks to models performing other tasks, including backdoor attacks on transfer learning [32], federated learning [33], self-supervised [34] and semi-supervised learning [35], as well as models for voice recognition [36] and natural language processing [37].

2.2 Existing Backdoor Defenses

The existing backdoor defense methods can be categorized into three types based on their focus: spatial domain-based, spectral domain-based, and semantic domain-based backdoor defenses.

Spatial Domain In image classification tasks, the spatial domain refers to the arrangement of pixels within each sample image. Backdoor defense methods that analyze from the spatial perspective attempt to detect backdoors directly without applying any transformations to the samples or the model.

Wang et al. [12] proposed the Neural Cleanse (NC) defense, which assumes the attacker will try to design the trigger as small as possible to draw less attention from human inspection. It therefore reverse-engineers each class to find a potential *trigger* and regards the anomalously small one as the actual backdoor trigger. Gao et al. [38] assumed that trigger features are usually strong, while benign features are otherwise fragile. Therefore, they proposed STRIP, which overlaps images and checks whether the model's predictions change. If not, the sample is likely poisoned. Selvaraju et al. [18] proposed Grad-CAM as a method to improve model explainability by highlighting which area of a sample the model focuses on during prediction. It has also been recognized as a defense against backdoor attacks and inspired many derivatives. For example, Doan et al. [39] proposed Februus, which uses model explainability to extract the area receiving the most attention during classification—usually the trigger area in a poisoned sample. They then cover this area and perform image reconstruction to restore a benign sample. Chou et al. [40] proposed SentiNet, which also uses Grad-CAM to locate the most important area in a sample and cover it to observe whether the model prediction changes.

Spectral Domain Spectral-based backdoor defense methods involve transforming image samples from the spatial domain to the frequency domain using techniques such as FFT (Fast Fourier Transform) or DCT (Discrete Cosine Transform). After transformation, these methods identify triggers and backdoors by detecting abnormal changes in frequencies and amplitudes caused by trigger insertion.

Hammoud et al. [41] proposed a defense that clusters frequency-domain response vectors of different samples (e.g., using k-means) to automatically distinguish poisoned samples from benign ones, without requiring prior knowledge of the trigger shape or target label. Zeng et al. [13] used the spectral anomalies caused by trigger injection to establish a trigger dataset, then used this dataset to train a classifier to detect poisoned samples. Fu et al. [42] proposed a method to detect malicious traffic. However, the proposed frequency-domain analysis can also be applied to detect anomalous patterns in deep learning models, including backdoor attacks. They leverage frequency-domain features to achieve efficient and robust real-time detection.

Semantic Domain The semantic domain refers to any space that can represent or maximize the features of a sample. This domain can include not only manually defined spaces but also those automatically discovered by the model during training. For instance, to classify samples more efficiently, the model often assigns one or more neurons to specific features. In this case, the set of neurons activated by a sample can be regarded as its representation in the model's abstract semantic domain. Based on these features, poisoned samples can be identified.

Liu et al. [14] proposed Fine-pruning, which assumes some neurons in the network are specifically responsible for the backdoor function and remain inactive when benign samples are input. By pruning these neurons, the backdoor can be eliminated. Tran et al. [43] proposed Spectral Signatures, which suggests that if a class contains both benign and poisoned samples, the poisoned samples will exhibit strong signals in activation states and can thus be detected by Singular Value Decomposition (SVD). Chen et al. [44] proposed Activation Clustering, which follows a similar assumption. Even if poisoned and benign samples in the target class are classified the same, their internal mechanisms differ. Clustering neuron activations of samples in a class reveals the target class with a much higher Silhouette Score. Liu et al. [45] proposed ABS, which analyzes neuron activations under varying stimulus intensities to locate poisoned neurons and reverse-engineer the trigger based on these neurons. Li et al. [46] proposed NAD, which uses a teacher model to fine-tune the backdoored model, forcing it to share hidden attention with the teacher model, thus mitigating the backdoor effect.

3 3S-attack

The aim of this research is to address the limitations of existing backdoor attacks. Existing backdoor attack schemes have considered the stealthiness of triggers and poisoned samples in the spatial domain (making the poisoned samples look similar to the corresponding benign samples), the spectral domain (ensuring the spectral map of poisoned samples has no obvious anomaly), and the semantic domain (ensuring the model activations for poisoned samples resemble those of benign samples). However, none of them aim to make backdoor attacks stealthy across all three domains simultaneously. Therefore, this paper focuses on designing a backdoor trigger that is stealthy in the spatial, spectral, and semantic domains. We name our attack *3S-attack*, as it satisfies all the above requirements. Furthermore, existing backdoor attacks that are stealthy in the semantic domain always

involve controlling or manipulating the model training process [17], which is unrealistic for attackers in most circumstances. To the best of our knowledge, this is also the first work to develop a backdoor attack scheme that is stealthy in the semantic domain without access to model parameters or the training process. Thus, this work expands the applicability of advanced backdoor attacks, posing a significant threat to the backdoor defense community.

3.1 Threat Model

In this work, we follow the most common assumptions adopted in previous studies [11, 15, 26, 16, 19, 21, 23, 47, 25, 27, 28, 30, 31, 35, 34].

Attacker's Capability The attacker can inject or alter a certain number of samples in the training dataset. For example, the attacker may generate poisoned samples and publish them online, waiting for victims to collect them as part of their training dataset; or the attacker may be a third-party data collection or labeling service provider who has more control over the victim's dataset.

However, like many previous works, we do not assume that the attacker has access to the model itself, such as the training process, model parameters, or loss function. This is because very few individuals or parties have access to a specific victim's model, such as MLaaS providers. Moreover, attacks conducted by such parties are highly traceable. Once a backdoor is discovered in the model, the victim can easily identify the misbehaving party and hold them accountable, which greatly limits the feasibility of such attacks.

Therefore, in this paper, we assume that the attacker has access to the training dataset but not to the model training process.

Attacker's Goal The attacker's goal is to successfully implant a backdoor into the target model via data poisoning. Specifically, the attacker designs a trigger, generates multiple poisoned samples using it, and relies on the victim to train a model with these samples. The backdoor attack should fulfill the following characteristics: feasibility (remains inactive on benign samples but causes misclassification to a target class when triggered), stealthiness (undetectable through human inspection across various domains), and defense resistance (resistant to defenses from different perspectives).

Problem Formulation We focus our work on deep neural networks (DNN) for image classification trained using supervised learning, as this is the most common setting for DNN usage and training in real-world applications.

Normally, a DNN model is trained with dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ with the following equation:

$$\theta^* = \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \ell(f_\theta(x_i), y_i), \tag{1}$$

where θ is the model parameters; $\ell(\cdot)$ is the cross-entropy loss; $f_\theta(\cdot)$ is the function of the model; x_i is a sample in the training dataset; y_i is the corresponding label, and $\frac{1}{N}\sum_{i=1}^{N}$ represents the average loss over all N training examples.

Now, the functionality of a DNN can be described as follows:

$$f_\theta(x_i) = t_i = y_i, \tag{2}$$

where $f_\theta(\cdot)$ is the function of the model; x_i is a sample submitted to the model; t_i is the model prediction class of this sample; and y_i is the label of this sample, perhaps the ground truth of this sample. This indicates the model should have a high prediction accuracy on benign samples, which is the basic requirement for a functional model.

The backdoor model on the other hand, is trained with a slightly modified dataset $\mathcal{D} = \mathcal{D}_{\text{clean}} \cup \mathcal{D}_{\text{poisoned}}$, where $\mathcal{D}_{\text{poisoned}} = \{(x_i^p, y_i^p)\}_{i=1}^{N_p}$ as the subset of poisoned samples following the same process in Eq. 1. In which case, the poisoned sample x_i^p is crafted via a data poisoning function $x_i^p = T(x_i)$ where $T(\cdot)$ donates the implanting the trigger into samples process and samples' corresponding class is changed from the original y_i to target class y_i^p. The backdoor model has not only the feature of Eq. 2, but also another function:

$$f_\theta(x_i^p) = t_i^p = y_i^p, \tag{3}$$

where x_i^p is the poisoned sample; t_i^p is the model prediction on the poisoned sample; and y_i^p is the target class. This indicates that the backdoor model can accurately classify benign samples, while misclassifying every poisoned sample into the target class. Since we assume no access to any model-related information, our attack design focuses on finding an optimal trigger that satisfies these goals.

3.2 Attack Method Intuition

The major challenge in designing a stealthy backdoor attack lies in making the trigger invisible in the semantic domain, which is an abstract space autonomously learned by the model and exhibits strong black-box characteristics. It is impossible to predict the shape of this space before training begins, let alone describe it accurately.

To address this challenge, the proposed 3S-attack adopts a strategy of *fighting magic with magic*. Theoretically, a fully trained model should focus on parts of an input image that best reflect the features associated with its label. For instance, if an image is labeled as a *cat*, the model should focus on the parts that reveal the presence of a cat (e.g., the cat's body), while ignoring irrelevant parts (e.g., the background). Hence, models trained on similar datasets are expected to focus on roughly the same regions when classifying the same sample.

Following this insight, the 3S-attack attempts to indirectly characterize the semantic domain and bypass the difficulty of describing it by leveraging a preliminary model to predict the semantic domain of the target model. Specifically, the attacker first trains a clean model and observes which neurons are activated when processing benign samples. No specific requirements are imposed on this clean model, as long as it achieves acceptable classification accuracy to allow Grad-CAM to compute saliency map for given samples. By analyzing the saliency map, the attacker identifies the parts of the image that are most important to the model. This information is then utilized to construct the trigger for the backdoor attack. The detailed procedures for trigger generation and injection are introduced below.

3.3 Trigger Extraction

The attacker first trains a preliminary model using a clean dataset. This model need not achieve optimal accuracy—only an acceptable performance level. The attacker then selects a target class and chooses one or more samples from this class to generate the trigger.

As shown in Figure 2, the attacker uses Grad-CAM to extract the regions the model relies on most when classifying the trigger samples

(saliency maps). These saliency maps are multiplied with the corresponding samples to produce *tailored samples*. Both the original trigger samples and the tailored samples are then transformed using the Discrete Cosine Transform.

$$F(u,v) = \frac{1}{4}\alpha(u)\alpha(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\cdot\cos\left(\frac{(2x+1)u\pi}{2M}\right)\cdot\cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

Where:

$f(x,y)$: pixel value at position (x,y) in the spatial domain

$F(u,v)$: DCT coefficient at frequency coordinates (u,v)

M, N : image size

$$\alpha(k) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } k = 0 \\ 1, & \text{if } k > 0 \end{cases} \text{ : normalization factor}$$

The attacker compares the magnitude of each frequency component in the resulting spectrograms. Frequencies with magnitude differences below a certain threshold are considered the key features that the model uses for prediction. These frequencies and their corresponding magnitudes are stored as the trigger.

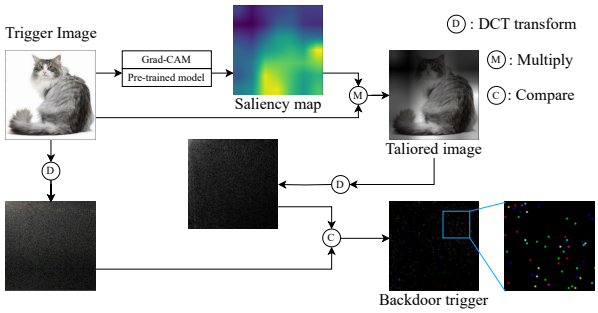

Figure 2: Pipeline for extracting a trigger from a benign sample in target class.

3.4 Poisoned Samples Generation

After obtaining the trigger, the next step is to inject it into samples to generate poisoned inputs. As shown in Figure 3, for a target sample, the attacker first applies DCT to obtain its spectral map. Then, based on a predefined parameter called the *Poison Distance Ratio*, the magnitudes of the trigger-identified frequencies in the target sample are adjusted toward the corresponding values in the trigger. After this adjustment, inverse DCT is applied to convert the sample back into the spatial domain.

However, preliminary experiments indicate that directly adding triggers in the spectral domain can result in unnatural artifacts in the spatial domain (see the upper half of Figure 4). Therefore, it is necessary to constrain pixel variations. Specifically, after inverse transformation, the modified sample is compared with the original in terms of pixel values. If the change in any pixel exceeds a certain threshold, the change is limited to that threshold. The same rule applies when pixel values exceed the data boundaries (e.g., 0–255 for uint8, or 0–1 for float data). Note that the pixel value change restriction step does not always take effect, as in most cases the pixel changes caused by trigger

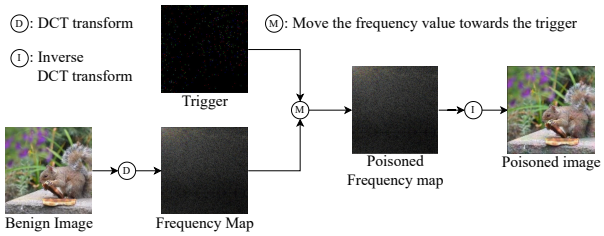

Figure 3: Process of embedding the trigger into benign samples to generate poisoned samples.

injection do not exceed the pixel change threshold. In other words, the pixel restriction serves merely as a safeguard in case the pixel changes become too large.

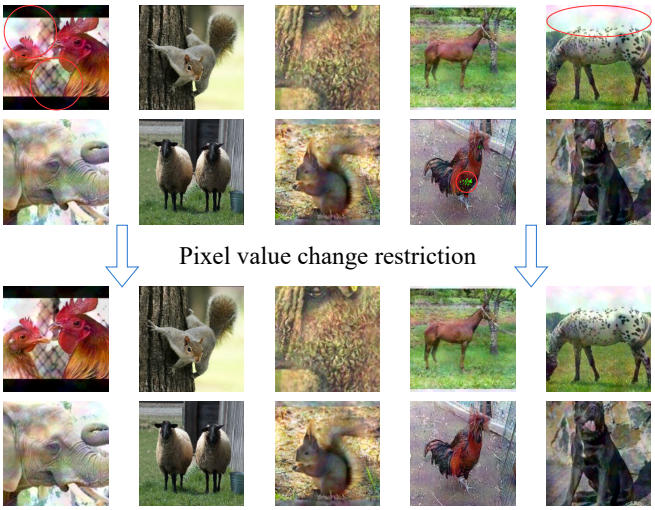

Figure 4: Pixel value change restriction on poisoned samples. Note that the red circles in the figure are solely used to highlight the unnatural artifacts in the samples; the circles themselves are not part of the poisoned samples.

3.5 3S-attack Overview

To summarize, the proposed 3S-attack follows a three-stage process that enables stealthy and effective backdoor injection by leveraging frequency-domain manipulation guided by semantic features. This design ensures that the poisoned samples remain stealthy across spatial, spectral, and semantic domains while evading multiple defense mechanisms. The entire procedure is illustrated in Algorithm 1, which includes the following components:

Trigger Extraction: A preliminary model is trained to generate Grad-CAM saliency maps for samples in the target class. These maps highlight class-relevant features. By comparing the DCT representations of the original and tailored images, a set of key frequency components is selected as the trigger pattern.

Poisoned Sample Generation: A subset of clean data is randomly selected, and their DCT coefficients are selectively modified at the trigger frequencies using linear interpolation between their original

values and those in the extracted trigger. The modified representations are then transformed back into the spatial domain via inverse DCT.

Pixel Value Restriction: To preserve visual imperceptibility, the pixel-level differences between each poisoned sample and its original are clipped to a specified threshold.

Algorithm 1 Trigger Extraction and Poisoned Sample Generation Algorithm of 3S-Attack.

Require: Clean dataset $\mathcal{D}$, target class c_t, frequency selection threshold δ, poison distance ratio α, pixel change restriction threshold τ

Ensure: Poisoned dataset $\mathcal{D}_{poisoned}$

▷ Step 1: Trigger extraction

Train model M on $\mathcal{D}$
Select sample(s) $x_{trig} \in c_t$
$S \leftarrow \text{Grad-CAM}(M(x_{trig}))$ ▷ Compute saliency maps
$\tilde{x}_{trig} \leftarrow S \odot x_{trig}$
$F_{ori} \leftarrow \text{DCT}(x_{trig})$
$F_{tailored} \leftarrow \text{DCT}(\tilde{x}_{trig})$
$\mathcal{F} \leftarrow \{f : |F_{ori}(f) - F_{tailored}(f)| < \delta\}$
Extract trigger: $\{(f, F_{ori}(f)) \mid f \in \mathcal{F}\}$

▷ Step 2: Poisoned sample generation

Sample subset $\mathcal{D}' \subset \mathcal{D}$
for all $x \in \mathcal{D}'$ **do**
  $F_x \leftarrow \text{DCT}(x)$
  for all $f \in \mathcal{F}$ **do**
    $F'_x(f) \leftarrow (1-\alpha) \cdot F_{ori}(f) + \alpha \cdot F_x(f)$
  end for
  $\hat{x} \leftarrow \text{IDCT}(F'_x)$

▷ Step 3: Pixel value change restriction

  for all pixel p in $\hat{x}$ **do**
    if $|\hat{x}(p) - x(p)| > \tau$ **then**
      $\hat{x}(p) \leftarrow x(p) + \text{sign}(\hat{x}(p) - x(p)) \cdot \tau$
    end if
    $\hat{x}(p) \leftarrow \text{clip}(\hat{x}(p), 0, 255)$
  end for
  Add $\hat{x}$ to $\mathcal{D}_{poisoned}$
end for
return $\mathcal{D}_{poisoned} \leftarrow \mathcal{D} \cup \mathcal{D}_{poisoned}$

4 Experiments

4.1 Experimental Setup

Environment All experiments were conducted on a server equipped with NVIDIA A100 Tensor Core GPUs and Intel® Xeon® Platinum 8570 CPUs, running Red Hat Enterprise Linux 8.10. Before running the experiments, we applied for 1 GPU, 10 nodes, and 40GB of memory. All experiments were performed using Python 3.11.0 and PyTorch 2.5.1+cu118. We used the Adam optimizer with a learning rate of 0.001, a batch size of 128, and trained the models for 50 epochs.

Datasets Backdoor attacks against DNNs have mainly focused on models for image classification tasks. Therefore, we selected datasets that are representative in the image classification field to demonstrate the generalizability of the 3S-attack across various scenarios.

Table 1: Details of each datasets.

Dataset	Input Size	#Train	#Test	Classes
MNIST	$28 \times 28 \times 1$	60000	10000	10
GTSRB	$32 \times 32 \times 3$	39209	12630	43
CIFAR-10	$32 \times 32 \times 3$	50000	10000	10
CIFAR-100	$32 \times 32 \times 3$	50000	10000	100
Animal-10	$128 \times 128 \times 3$	23679	2500	10

We strategically selected MNIST, GTSRB, CIFAR-10, CIFAR-100, and Animal-10 to comprehensively evaluate our attack's feasibility, stealthiness, and resistance to defenses. MNIST [48] is a simple handwritten digit dataset containing grayscale images of numbers from 0 to 9. GTSRB [49] stands for the German Traffic Sign Recognition Benchmark and contains 43 types of traffic signs. Each sample is an RGB image clipped from real-world photographs. CIFAR-10 [50] contains images categorized into 10 classes, including objects such as *airplane* and animals such as *cat*, with each sample being an RGB image. CIFAR-100 [50] includes 100 image classes covering a wide range of objects, buildings, creatures, and scenes, simulating high inter-class similarity. Animal-10 [51] is a dataset for testing models on high-resolution images. It contains 10 animal classes with each sample being a high-resolution RGB image. Table 1 shows the detailed statistics of each dataset.

The selection of these datasets was based on key considerations to ensure a thorough and balanced evaluation. They are representative of a wide range of classification challenges, from simple tasks such as digit recognition to complex real-world applications. They also span a range of computational demands, allowing us to assess performance under different resource constraints. Moreover, these datasets serve as historical benchmarks in the field, enabling fair comparisons with state-of-the-art models and existing defenses. Their structured diversity—encompassing handwritten digits, traffic signs, objects, animals, and high-resolution images—ensures a robust evaluation of 3S-attack across varied conditions.

Models We employ different neural network architectures tailored to the complexity and characteristics of each dataset to ensure a meaningful and rigorous evaluation. The selected models include lightweight, standard, and high-capacity architectures, allowing analysis of the attack's effectiveness across varying levels of model complexity and feature extraction capabilities.

For MNIST, we utilize both a custom small model and LeNet-5, as this dataset consists of low-resolution grayscale images of digits, requiring relatively simple architectures. LeNet-5 is a classical CNN benchmark for such tasks. The custom model provides flexibility to evaluate the attack's feasibility under different architectures.

For GTSRB and CIFAR-10, we use VGG-11 and ResNet-18 to study the impact of model depth and feature extraction strategy. VGG-11 offers a simple stacked architecture, while ResNet-18, with its residual connections, is designed to address vanishing gradients and enhance feature reuse.

For CIFAR-100, we use WideResNet (WRN), a ResNet variant with wider layers, offering greater feature representation capacity, suitable for this fine-grained classification task.

For Animal-10, we adopt ResNet-18 due to its balance between capacity and efficiency, and because it is widely used in high-resolution

classification tasks.

Metrics 1: ASR Attack Success Rate (ASR) measures the effectiveness of a backdoor attack by quantifying the probability that a model misclassify poisoned samples as the target class when the trigger is present. It is defined as:

$$ASR = \frac{|\{(x_i^p,\ y_i^p) \in \mathcal{D}_{poisoned} | f_\theta(x_i^p) = y_i^p\}|}{|\{(x_i^p,\ y_i^p) \in \mathcal{D}_{poisoned}\}|}, \quad (4)$$

Where:

(x_i^p, y_i^p) : a poisoned input sample and its associated target label
$\mathcal{D}_{\text{poisoned}}$: a dataset that only contain poisoned samples
$f_\theta(x_i^p)$: the model's prediction for poisoned input x_i^p
y_i^p : the target class chosen by the attacker
$|\cdot|$: cardinality, i.e., the number of elements in the set

A higher ASR indicates a more effective backdoor attack, as it implies consistent misclassifications caused by the trigger. Vice versa, a low ASR suggests the attack is weak or susceptible to defenses.

Metrics 2: PSNR Peak Signal-to-Noise Ratio (PSNR) evaluates the stealthiness of a backdoor trigger in pixel level by measuring the pixel level similarity between the original and poisoned samples. It is defined as:

$$\text{PSNR} = 10 \cdot \log_{10}\left(\frac{\text{MAX}^2}{\text{MSE}}\right), \quad (5)$$

Where:

MAX : maximum possible pixel value of the image
MSE : Mean Squared Error between the images
$x(i,j), y(i,j)$: pixel intensities at position (i,j)
m, n : image height and width, respectively

A high PSNR value indicates that the poisoned image closely resembles the original, making the backdoor attack more stealthy. Conversely, a low PSNR suggests visible distortion, increasing the likelihood of detection.

Metrics 3: SSIM Structural Similarity Index Measure (SSIM) assesses the perceptual similarity between the original and poisoned images in global level, considering not only pixel-wise differences but also structural information. It is given by:

$$\text{SSIM}(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (6)$$

Where:

x, y : input images being compared (e.g., original and poisoned)
μ_x, μ_y : mean intensity (average pixel value) of images x and y
σ_x^2, σ_y^2 : variance (contrast) of images x and y
σ_{xy} : covariance between x and y
C_1, C_2 : small constants to stabilize the division

A high SSIM value (close to 1) indicates strong structural similarity between the two images, suggesting that the perturbation (e.g., trigger) is perceptually subtle. Compared to PSNR, SSIM better aligns with human visual perception by incorporating luminance, contrast, and structural components.

Together, these three metrics evaluate a backdoor attack's effectiveness (ASR) and stealthiness (PSNR and SSIM) from complementary perspectives. ASR (Attack Success Rate) evaluates the reliability of the attack by measuring how effectively the model misclassifies poisoned samples into the attacker-specified target class. PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index Measure) both assess the imperceptibility of the trigger, but from different perspectives. PSNR quantifies pixel-level distortions by measuring the absolute differences in intensity between original and poisoned images, offering a low-level view of visual changes. In contrast, SSIM captures the perceptual quality of the image by accounting for luminance, contrast, and structural information, thereby providing a more holistic and human-aligned assessment of visual similarity.

4.2 Attack Performance Evaluation

Baseline Attack We selected several baseline backdoor attack methods that employ different trigger and poisoned sample generation algorithms to compare with 3S-attack. Specifically, we chose Wanet [15], Bppattack [26], ISSBA [47], FIBA [16], and BadNets [11]. Wanet [15] defines a warping field as the trigger and applies it to benign samples, it acts in the spatial domain. Bppattack [26] uses quantization and dithering as the trigger mechanisms, it acts in the spatial domain. ISSBA [47] trains an encoder-decoder pair initially designed for steganography to embed hidden triggers, it acts in the semantic domain. FIBA [16] selects the central frequencies of a benign sample as trigger and replaces that of other samples to generate poisoned inputs, it acts in the spectral domain. BadNets [11] serves as a standard baseline and is used to evaluate the effectiveness of various defenses, it acts in the spatial domain.

Attack Performance We compare the proposed 3S-attack with other backdoor attacks, and Table 2 demonstrates the comparative performance of each backdoor attack across the above mentioned datasets. Specifically, the 3S-attack attains a consistently high ASR across each datasets, showing that it is achieving a acceptable attack feasibility. More importantly, it demonstrates remarkably high PSNR and SSIM scores across all datasets—for instance, a PSNR of 35.65 and SSIM of 0.9690 on CIFAR-10, which surpass those of Wanet Wanet [15] (29.95 / 0.7735) and Bppattack [26] (20.06 / 0.9233). These results indicate that the perturbations introduced by 3S-attack are not only effective but also imperceptible. Therefore, compared with baseline attacks which sacrifice trigger stealthiness for ASR, 3S-attack offers a better trade-off between effectiveness and imperceptibility. Besides, 3S-attack is having a constant performance across different dataset, indicating its strong generalization capability. Note that in Table 2, the results of BppAttack on MNIST are omitted because BppAttack is incompatible with the data characteristics of MNIST. Specifically, most pixel values in MNIST are either 0 (the minimum) or 255 (the maximum), resulting in a highly saturated dataset. When BppAttack is applied, it often produces pixel values that exceed these limits. Due to value clipping, any modifications that fall outside the valid pixel range are suppressed, rendering the inserted triggers ineffective. As a result, BppAttack consistently fails to generate valid poisoned samples on MNIST.

Table 2: ASR, PSNR, and SSIM value of different attacks in spatial domain. Note that the ASR is in percentage format.

	3S-attack			ISSBA [47]			Wanet [15]			Bppattack [26]			FIBA [16]		
Datasets	ASR	PSNR	SSIM	ASR	PSNR	SSIM	ASR	PSNR	SSIM	ASR	PSNR	SSIM	ASR	PSNR	SSIM
MNIST	96.47	**46.01**	**0.943**	**99.10**	39.22	0.892	97.43	34.13	0.639	-	-	-	70.68	23.93	0.679
GTSRB	94.12	**32.78**	**0.979**	93.71	19.03	0.653	**98.31**	31.22	0.759	95.29	24.61	0.943	79.05	14.62	0.559
CIFAR10	89.29	**35.65**	**0.969**	77.23	23.51	0.852	**93.36**	29.95	0.773	91.32	20.06	0.923	65.85	15.50	0.710
CIFAR100	92.38	**31.68**	**0.946**	86.42	22.79	0.851	**93.06**	30.69	0.858	85.94	20.12	0.927	75.48	15.87	0.770
Animal10	97.42	**30.83**	**0.962**	**99.87**	26.6	0.840	93.88	29.59	0.452	92.44	23.28	0.966	58.72	15.69	0.754

4.3 Parameters

In 3S-attack, there are multiple parameters and thresholds that can affect the performance of the attack. Among them, the most important parameters are:

1. Poison rate: The poison rate is the most intuitive way to demonstrate the performance of a backdoor attack, as is well known that with less poisoned samples needed for successfully embedding a backdoor, comes a more effective backdoor attack.

2. Frequency selection threshold: After transformed target image and tailored image to frequency domain, each frequencies at the same location in both frequency map is compared. If their numerical similarity percentage is above the frequency selection threshold, then this frequency and the corresponding value is selected as part of the trigger.

3. Poison distance ratio: When injecting the trigger into samples, the amplitude of trigger frequencies in benign sample will move towards the value in trigger and poison distance ratio is the extent this amplitude will moves.

4. Pixel change restriction threshold: This parameter controls to what extent the tolerance is on pixel value change on poisoned samples. With a higher threshold, each pixel can change more when modifying the frequency map of a sample. However, with a lower threshold, less change is allowed on each pixel, which could lead to a more stealthy change to samples, but also may cause the trigger to be too weak to embed the backdoor.

Next, we evaluate how these parameters affect 3S-attack performance. Figure 5, illustrate the effects on ASR for CIFAR-10, CIFAR-100, and Animal-10, respectively. Generally, ASR increases as the proportion of trigger components in the dataset increases, which is intuitive—higher poisoned intensity leads to higher ASR. And with the increase of number of poisoned samples (sub-figure a), frequency selection threshold (sub-figure b), poison distance ratio (sub-figure c), and pixel change restriction threshold (sub-figure d), the proportion of trigger components in the dataset increases.

For CIFAR-10, we see that a very high frequency selection threshold can reduce ASR. A possible explanation is that selecting too many frequencies as triggers causes the attachment of the trigger to have an excessive impact on samples in the spatial domain. This results in significant pixel changes that exceed the preset threshold. Consequently, when pixel changes are subsequently constrained, the magnitudes of many frequencies in the spectral map are altered in reverse, rendering the trigger in the frequency domain ineffective.

Overall, as illustrated in the datas (Fig 5 and Tab 2), the 3S-attack exhibits strong stealthiness and robustness. Even when the proportion of trigger components in the dataset is low—corresponding to conservative parameter settings—it consistently achieves a satisfactory ASR. Furthermore, it maintains a relatively high ASR across a wide range of parameter variations. These results suggest that effective attack performance can be attained with high probability, even in the absence of detailed knowledge about the target model or dataset. This highlights the practicality of the 3S-attack, as its parameters can be configured based on general intuition or prior experience rather than precise model-specific tuning.

4.4 Defense Resistance

The ability to evade existing defenses is a fundamental requirement for modern backdoor attacks, as any attack lacking this capability is likely to be detected and mitigated. The 3S-attack is specifically designed to remain imperceptible across the spatial, spectral, and semantic domains. Accordingly, we selected a comprehensive set of defense methods that target these three domains to evaluate the defense resistance of 3S-attack. In particular, we consider the following representative defenses: STRIP [38], Neural Cleanse (NC) [12], Activation Clustering (AC) [44], Grad-CAM [18], Fine-Pruning [14], and Frequency-based Trigger Detection (FTD) [13] where STRIP, NC, and Grad-CAM are based on spatial domain; FTD is based on spectral domain; and FP and AC is based on semantic domain. The remainder of this section presents the evaluation of 3S-attack against each of these defenses in detail.

STRIP The core idea behind STRIP is that backdoor triggers are typically designed to be highly robust in order to ensure a high attack success rate, whereas benign features in input samples tend to be more fragile and susceptible to disruption. Consequently, when a sample is perturbed by blending it with other inputs, the benign features are more likely to be compromised, leading to changes in the model's prediction. In contrast, the backdoor trigger remains effective even when mixed with other inputs, often still causing the model to misclassify the sample into the target class. By analyzing the entropy distribution of model predictions under such perturbations, STRIP distinguishes between benign and poisoned samples, allowing for the establishment of a threshold to detect and filter out potential backdoors.

Figure 6 presents the performance of STRIP against BadNets [11] and the proposed 3S-attack on the GTSRB and Animal10 datasets. The results show that STRIP is effective in identifying poisoned samples in BadNets, as their distribution (orange) significantly diverges from that of benign samples (blue). However, in the case of 3S-attack, the distribution of poisoned samples (green) closely resembles that of benign samples, making them indistinguishable. As a result, no reliable threshold can be set to effectively separate poisoned samples introduced by 3S-attack, allowing it to successfully evade detection by STRIP.

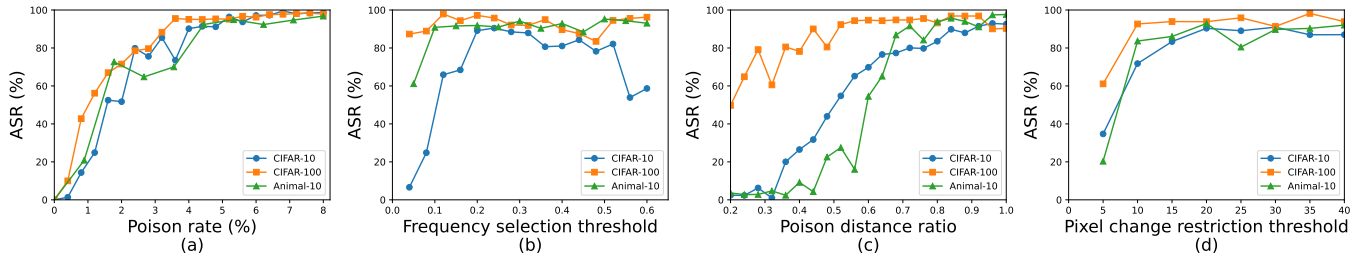

Figure 5: The effect of (a) poison rate, (b) frequency selection threshold, (c) poison distance ratio, and (d) pixel change restriction threshold on ASR, evaluated on three datasets: CIFAR-10, CIFAR-100, and Animal-10. Increasing any of these parameters generally leads to higher ASR, indicating stronger attack effectiveness.

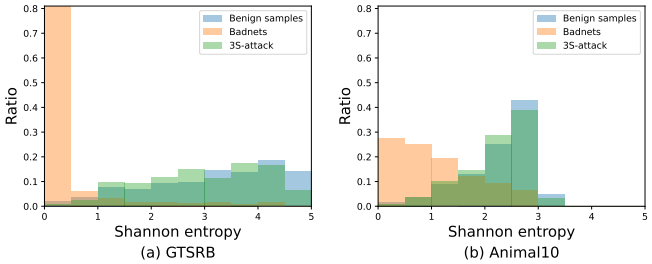

Figure 6: Experimental results of STRIP against Badnets and 3S-attack on GTSRB and Animal10 datasets.

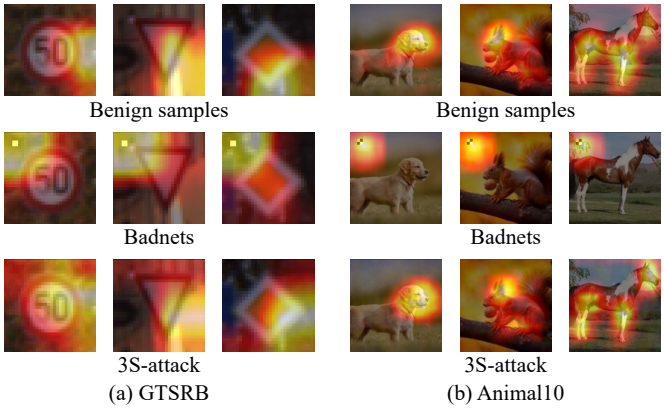

Figure 7: Experimental results of Grad-CAM against Badnets and 3S-attack on GTSRB and Animal10 datasets.

Grad-CAM Grad-CAM is employed not only as a component in generating the 3S trigger but also as an established method for backdoor defense. When used defensively, Grad-CAM produces a heatmap (saliency map) that highlights the regions of an input sample to which the model pays the most attention during classification. In benign samples, the highlighted area typically corresponds to the primary object or key discriminative features. In contrast, for poisoned samples, the highlighted region often corresponds to the location of the trigger.

Figure 7 illustrates saliency maps for benign samples (top), poisoned samples from BadNets (middle), and poisoned samples from the 3S-attack (bottom). For benign samples, the model's attention is correctly concentrated on the main features or objects within the image. However, in poisoned samples generated by BadNets, the model's focus is predominantly on the trigger region, regardless of the true label or semantic content of the image. In contrast, the model's behavior on poisoned samples from the 3S-attack closely resembles its behavior on benign samples, with attention distributed over the primary semantic regions. This is because the 3S-attack trigger is embedded using features already associated with the benign class, resulting in no specific spatial region being consistently highlighted as the trigger area.

Fine-pruning Fine-pruning (FP) is based on the assumption that certain neurons in a backdoored model are specifically responsible for recognizing the presence of a trigger. These backdoor-associated neurons are typically inactive when processing benign samples and are only activated by poisoned inputs. Therefore, if a clean dataset that shares the same distribution as the training data is available, it can be fed into the model to monitor neuron activation states. Neurons that remain consistently inactive across the benign dataset are then considered likely to be backdoor-related and are subsequently pruned or de-weighted.

Figure 8 presents the results of applying the FP defense to the 3S-attack on the GTSRB and Animal10 datasets. It is evident that as the pruning rate increases, the benign accuracy (BA) declines more rapidly and earlier than the attack success rate (ASR). This indicates that there is no effective pruning threshold at which ASR is substantially reduced without also significantly degrading the model's performance on benign samples. A likely explanation is that the 3S-attack embeds the trigger using complex, distributed features that engage a wide range of neurons. As a result, neurons responsible for recognizing benign features and those involved in recognizing the trigger may overlap. This makes it difficult to isolate and remove backdoor-specific neurons without simultaneously impairing the model's normal functionality.

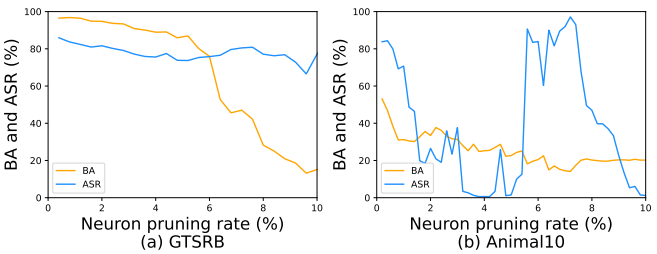

Figure 8: Experimental results of FP defense against 3S-attack on GTSRB and Animal10 datasets.

Frequency based Defense The core idea behind frequency-based defense (FTD) is that most backdoor triggers are designed in the spatial domain, often without careful consideration of their stealthiness in the spectral domain. As a result, when such triggers are analyzed in the frequency domain, they often leave distinct and detectable patterns in the frequency spectrum. However, although these patterns may appear visually conspicuous, it is challenging to design a rule-based algorithm that can reliably distinguish between benign and poisoned spectral signatures. To address this, FTD employs a neural network-based approach. Specifically, the defender collects a diverse set of known trigger patterns and uses them to generate poisoned samples. A binary classifier is then trained to differentiate between benign and poisoned samples based on their spectral characteristics.

Table 3 shows that FTD performs well in detecting certain types of backdoor attacks and their corresponding triggers. Notably, even when trained on a limited variety of trigger patterns, the FTD detector demonstrates some generalization ability and can successfully detect previously unseen trigger types. However, its effectiveness diminishes when facing attacks like Wanet [15] and the proposed 3S-attack. This is because the triggers in these attacks exhibit significantly different frequency-domain characteristics compared to those used in the training set. In particular, the 3S-attack modifies only a very small subset of frequency components, making the resulting spectral changes too subtle for the detector to reliably distinguish from benign samples. As a result, the FTD classifier fails to recognize the poisoned samples generated by 3S-attack as anomalous.

Attack methods	Detection Rate (%) GTSRB	Animal10
Benign samples	98.54	100
3S-attack	**1.46**	**0.98**
ISSBA	100	99.16
Wanet	6.11	4.36
Bppattack	98.87	99.44
FIBA	99.98	98.76
Badnets	100	99.08

Table 3: Experimental results of FTD defense method against multiple backdoor attack schemes on GTSRB and Animal10 datasets.

However, the 3S-attack also have its vulnerability that can be detected by certain type of backdoor defense like Neural Cleanse and Activation Clustering.

Neural Cleanse The core idea behind Neural Cleanse (NC) is based on the observation that attackers typically aim to design triggers as small and inconspicuous as possible. Moreover, backdoor-ed models often rely on a few key pixels from the trigger pattern to cause misclassifications. As a result, for the target class, it is usually possible to identify a small trigger pattern that, when attached to a wide range of benign inputs, consistently causes misclassification into that class. In contrast, for clean (non-target) classes, any synthesized *trigger* that causes benign samples to be misclassified into those classes tends to be much larger, as there is no actual backdoor associated with them. By reverse-engineering potential *triggers* for all classes and comparing their sizes, NC identifies the class with an abnormally small trigger as the likely backdoor target.

Figure 9 presents the anomaly index for each class in the NC defense applied to a 3S-attack where the target class is *7*. Subfigure (a) shows the results on the GTSRB dataset, where class *7* exhibits a significantly higher anomaly index, well above the detection threshold, indicating that the 3S-attack is effectively detected. However, in subfigure (b), based on the Animal10 dataset, the anomaly index of class *7* is 1.73—still relatively high but below the threshold. Moreover, another clean class also has a comparable anomaly index of 1.65. These results suggest that while NC is effective in detecting the 3S-attack under certain conditions, its reliability is not guaranteed across all settings. Due to the black-box nature of DNNs, the underlying reasons for this inconsistency are difficult to pinpoint. One possible explanation is that, in some cases, the model learns to associate certain subtle, recurring pixel patterns—shared across all poisoned samples—as the effective trigger, thereby enabling successful reverse engineering by NC.

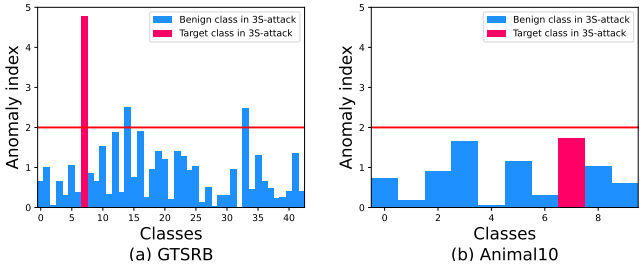

Figure 9: Experimental results of Neural Cleanse against 3S-attack on GTSRB and Animal10 datasets.

Activation Clustering The idea behind Activation Clustering (AC) is similar to that of Fine-Pruning, in that certain neurons in a backdoored model—particularly those in the fully connected layers—are specifically responsible for recognizing the presence of a trigger. As a result, although poisoned and benign samples from the target class may yield the same prediction, the internal mechanisms differ, as they activate different subsets of neurons. Based on this observation, for each class in the model, one can collect neuron activation patterns and apply clustering analysis. If the activations naturally separate into two distinct clusters, it is likely that the class is a backdoor target; otherwise, the class is considered benign.

Figure 10 illustrates that AC is effective against the 3S-attack across different datasets, as the Silhouette scores of the target class are consistently higher than those of benign classes. This may be attributed to the fact that, although 3S-attack is designed to make poisoned samples activate similar neurons as benign ones, the internal optimization process of the target model remains a black box and is beyond the attacker's control. Consequently, some neurons may still be implicitly assigned the task of recognizing trigger-specific patterns. This results in distinguishable activation differences within the target class, making it more prone to clustering. These findings suggest that AC is a particularly strong defense and that designing a backdoor attack capable of evading AC without access to the model training process remains an extremely challenging task.

5 Discussion

In this section, we analyze the key findings from the experiments, compare 3S-attack with existing works, identify limitations, and dis-

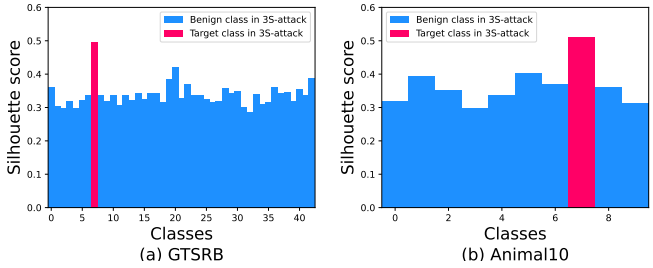

Figure 10: Experimental results of Activation Clustering against 3S-attack on GTSRB and Animal10 datasets.

cuss potential future directions.

Core Properties The experimental results demonstrate that 3S-attack is a feasible, stealthy, robust, and defense-resistant backdoor attack. It achieves consistently high ASR across datasets of varying complexity and resolution, including MNIST, GTSRB, CIFAR-10/100, and Animal-10, confirming its general feasibility. Meanwhile, the attack induces only minimal perceptual distortion, as evidenced by high PSNR and SSIM scores—often exceeding all baseline methods. This validates its spatial and perceptual stealthiness.

Hyperparameter and Model Robustness The 3S-attack remains stable across a wide range of parameters, including poison rate, frequency threshold, poison distance ratio, and pixel-level restriction. Even under conservative configurations, 3S-attack retains high effectiveness, showing robustness to hyperparameter variations. Moreover, it generalizes well across different model architectures, from simple CNNs to deep residual networks, further enhancing its applicability.

Defense Resistance Several defense mechanisms are rendered ineffective against 3S-attack. STRIP fails to detect poisoned samples due to overlapping entropy distributions between benign and poisoned samples are close. Grad-CAM-based detection is also evaded because Grad-CAM consistently highlights natural areas, even in poisoned samples. As a result, not only Grad-CAM but also its derivative defenses—such as saliency-based trigger localization—are effectively bypassed.

Failure of FTD FTD is designed to detect spectral anomalies but fails against 3S-attack. As shown in Figure 2, the trigger typically occupies only 1%–5% of the frequency map and lacks any structured or localized pattern. This seemingly randomness prevents the FTD classifier, trained on known triggers with regular frequency characteristics, from generalizing to 3S-attack. Consequently, FTD consistently misclassifies 3S-poisoned samples as benign.

Partial Detection by NC and AC Despite its stealth, 3S-attack remains partially detectable by Neural Cleanse (NC) and Activation Clustering (AC). NC succeeds in datasets like GTSRB, where class patterns are constrained, but fails on Animal-10 due to semantic complexity. AC is more robust: although 3S-attack aligns poisoned inputs with benign attention maps, it cannot fully eliminate discrepancies in deep-layer activations. These latent differences remain clusterable, suggesting that 3S-attack does not yet achieve complete semantic stealth.

Contributions and Impact This work is the first to propose a backdoor attack that is simultaneously stealthy in spatial, spectral, and semantic domains. Furthermore, it achieves semantic stealthiness without requiring access to the model training process—an important advancement for practical black-box attacks. These findings imply that backdoor attacks can remain effective even under strong stealth constraints, underscoring the considerable potential for advancement in the design of both backdoor attacks and corresponding defenses.

Limitations and Future Work The key limitation of 3S-attack lies in its incomplete stealth at the feature (semantic) level. Specifically, Activation Clustering can still detect subtle activation differences between benign and poisoned samples. Enhancing semantic invisibility without access to model internals remains a difficult but essential direction. Future work may explore: (1) adaptive frequency selection strategies, (2) activation-aligned poisoning to evade AC, and (3) extending the attack to more complex modalities such as video, text, and multimodal learning.

Summary 3S-attack demonstrates that multi-domain stealth is both achievable and effective. It exposes critical vulnerabilities in current AI systems and motivates the design of more advanced defense strategies.

6 Conclusion

In this paper, we proposed 3S-Attack, a novel backdoor attack that achieves stealthiness across spatial, spectral, and semantic domains. The attack constructs a triple-stealthy trigger by extracting class-relevant features using a preliminary model and Grad-CAM, followed by frequency-domain embedding and pixel-level constraint. Unlike prior works, 3S-Attack requires no access to the victim model or its training process, making it applicable to realistic threat scenarios. Extensive experiments demonstrate that our method not only maintains high attack success rates, but also achieves superior imperceptibility, effectively evading multiple existing defenses.

However, our current approach—while not relying on training-time access—does not fully guarantee invisibility in the model's internal feature space (semantic domain). This highlights a fundamental challenge for black-box backdoor attacks: achieving semantic-level stealth without participating in the training process. Addressing this limitation represents an important direction for future research.

References

[1] Xudong Pan, Mi Zhang, Beina Sheng, Jiaming Zhu, and Min Yang. Hidden trigger backdoor attack on nlp models via linguistic style manipulation. In *31st USENIX Security Symposium (USENIX Security)*, pages 3611–3628. USENIX Association, 2022.

[2] Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security)*, pages 5253–5270. USENIX Association, 2023.

[3] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: A query-efficient black-

box adversarial attack via random search. In *European Conference on Computer Vision (ECCV)*, pages 484–501. Springer, 2020.

[4] Junjian Li, Honglong Chen, Zhichen Ni, Yudong Gao, Weifeng Liu, and Nan Jiang. Image-scaling attack on image signal processing pipelines in deep neural networks based outdoor vision applications. *IEEE Transactions on Consumer Electronics*, 2024.

[5] Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In *2018 IEEE Symposium on Security and Privacy (S&P)*, pages 19–35. IEEE, 2018.

[6] Vale Tolpegin, Stacey Truex, Mehmet Emre Gursoy, and Ling Liu. Data poisoning attacks against federated learning systems. In *Computer Security–ESORICS 2020: 25th European Symposium on Research in Computer Security, ESORICS 2020, Guildford, UK, September 14–18, 2020, Proceedings, Part I 25 (ESORICS)*, pages 480–501. Springer, 2020.

[7] Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S Yu, and Xuyun Zhang. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)*, 54(11s):1–37, 2022.

[8] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 253–261. IEEE, 2020.

[9] Benjamin Zi Hao Zhao, Aviral Agrawal, Catisha Coburn, Hassan Jameel Asghar, Raghav Bhaskar, Mohamed Ali Kaafar, Darren Webb, and Peter Dickinson. On the (in) feasibility of attribute inference attacks on machine learning models. In *2021 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 232–251. IEEE, 2021.

[10] Soham Pal, Yash Gupta, Aditya Shukla, Aditya Kanade, Shirish Shevade, and Vinod Ganapathy. Activethief: Model extraction using active learning and unannotated public data. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 34, pages 865–872. AAAI Press, 2020.

[11] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019.

[12] 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 *2019 IEEE Symposium on Security and Privacy (S&P)*, pages 707–723. IEEE, 2019.

[13] Yi Zeng, Won Park, Z Morley Mao, and Ruoxi Jia. Rethinking the backdoor attacks' triggers: A frequency perspective. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16473–16481. IEEE, 2021.

[14] 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*, pages 273–294. Springer, 2018.

[15] Anh Nguyen and Anh Tran. Wanet–imperceptible warping-based backdoor attack. *arXiv Preprint arXiv:2102.10369*, 2021.

[16] Yu Feng, Benteng Ma, Jing Zhang, Shanshan Zhao, Yong Xia, and Dacheng Tao. Fiba: Frequency-injection based backdoor attack in medical image analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20876–20885. IEEE, 2022.

[17] Nan Zhong, Zhenxing Qian, and Xinpeng Zhang. Imperceptible backdoor attack: From input space to feature representation. *arXiv Preprint arXiv:2205.03190*, 2022.

[18] 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 *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 618–626. IEEE, 2017.

[19] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv Preprint arXiv:1712.05526*, 2017.

[20] Mauro Barni, Kassem Kallas, and Benedetta Tondi. A new backdoor attack in cnns by training set corruption without label poisoning. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 101–105. IEEE, 2019.

[21] Giulio Lovisotto, Simon Eberz, and Ivan Martinovic. Biometric backdoors: A poisoning attack against unsupervised template updating. In *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 184–197. IEEE, 2020.

[22] Haiqing Liu, Daoxing Li, and Yuancheng Li. Poisonous label attack: Black-box data poisoning attack with enhanced conditional dcgan. *Neural Processing Letters*, 53(6):4117–4142, 2021.

[23] Mingfu Xue, Can He, Jian Wang, and Weiqiang Liu. One-to-n & n-to-one: Two advanced backdoor attacks against deep learning models. *IEEE Transactions on Dependable and Secure Computing*, 19(3):1562–1578, 2020.

[24] Haoti Zhong, Cong Liao, Anna Cinzia Squicciarini, Sencun Zhu, and David Miller. Backdoor embedding in convolutional neural network models via invisible perturbation. In *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy (CODASPY)*, pages 97–108. Association for Computing Machinery, 2020.

[25] Minhui Zou, Yang Shi, Chengliang Wang, Fangyu Li, WenZhan Song, and Yu Wang. Potrojan: Powerful neural-level trojan designs in deep learning models. *arXiv Preprint arXiv:1802.03043*, 2018.

[26] Zhenting Wang, Juan Zhai, and Shiqing Ma. Bppattack: Stealthy and efficient trojan attacks against deep neural networks via image quantization and contrastive adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15074–15084. IEEE, 2022.

[27] Yi Yu, Yufei Wang, Wenhan Yang, Shijian Lu, Yap-Peng Tan, and Alex C Kot. Backdoor attacks against deep image compression via adaptive frequency trigger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12250–12259. IEEE, 2023.

[28] Yudong Gao, Honglong Chen, Peng Sun, Junjian Li, Anqing Zhang, Zhibo Wang, and Weifeng Liu. A dual stealthy backdoor: From both spatial and frequency perspectives. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pages 1851–1859. AAAI Press, 2024.

[29] Ruixiang Tang, Mengnan Du, Ninghao Liu, Fan Yang, and Xia Hu. An embarrassingly simple approach for trojan attack in deep neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, pages 218–228. Association for Computing Machinery, 2020.

[30] Khoa Doan, Yingjie Lao, Weijie Zhao, and Ping Li. Lira: Learnable, imperceptible and robust backdoor attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11966–11976. IEEE, 2021.

[31] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning attack on neural networks. In *25th Annual Network and Distributed System Security Symposium (NDSS)*. Unknown Publisher, 2018.

[32] Yuanshun Yao, Huiying Li, Haitao Zheng, and Ben Y Zhao. Latent backdoor attacks on deep neural networks. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 2041–2055. Association for Computing Machinery, 2019.

[33] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International Conference on Artificial Intelligence and Statistics*, pages 2938–2948. Unknown Publisher, 2020.

[34] Aniruddha Saha, Ajinkya Tejankar, Soroush Abbasi Koohpayegani, and Hamed Pirsiavash. Backdoor attacks on self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13337–13346. IEEE, 2022.

[35] Zhicong Yan, Gaolei Li, Yuan TIan, Jun Wu, Shenghong Li, Mingzhe Chen, and H Vincent Poor. Dehib: Deep hidden backdoor attack on semi-supervised learning via adversarial perturbation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pages 10585–10593. AAAI Press, 2021.

[36] Cong Shi, Tianfang Zhang, Zhuohang Li, Huy Phan, Tianming Zhao, Yan Wang, Jian Liu, Bo Yuan, and Yingying Chen. Audio-domain position-independent backdoor attack via unnoticeable triggers. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, pages 583–595. Unknown Publisher, 2022.

[37] Pengzhou Cheng, Zongru Wu, Wei Du, Haodong Zhao, Wei Lu, and Gongshen Liu. Backdoor attacks and countermeasures in natural language processing models: A comprehensive security review. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

[38] Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C. Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference (ACSAC)*, pages 113–125. Association for Computing Machinery, 2019.

[39] Bao Gia Doan, Ehsan Abbasnejad, and Damith C. Ranasinghe. Februus: Input purification defense against trojan attacks on deep neural network systems. In *Proceedings of the 36th Annual Computer Security Applications Conference (ACSAC)*, pages 897–912. Association for Computing Machinery, 2020.

[40] Edward Chou, Florian Tramer, and Giancarlo Pellegrino. Sentinet: Detecting localized universal attacks against deep learning systems. In *2020 IEEE Security and Privacy Workshops (SPW)*, pages 48–54. IEEE, 2020.

[41] Hasan Abed Al Kader Hammoud, Adel Bibi, Philip HS Torr, and Bernard Ghanem. Don't freak out: A frequency-inspired approach to detecting backdoor poisoned samples in dnns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2338–2345. IEEE, 2023.

[42] Chuanpu Fu, Qi Li, Meng Shen, and Ke Xu. Realtime robust malicious traffic detection via frequency domain analysis. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 3431–3446. Association for Computing Machinery, 2021.

[43] Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. *Advances in Neural Information Processing Systems*, 31, 2018.

[44] Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv Preprint arXiv:1811.03728*, 2018.

[45] Yingqi Liu, Wen-Chuan Lee, Guanhong Tao, Shiqing Ma, Yousra Aafer, and Xiangyu Zhang. Abs: Scanning neural networks for back-doors by artificial brain stimulation. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 1265–1282. Association for Computing Machinery, 2019.

[46] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Neural attention distillation: Erasing backdoor triggers from deep neural networks. *arXiv Preprint arXiv:2101.05930*, 2021.

[47] Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. Invisible backdoor attack with sample-specific triggers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16463–16472. IEEE, 2021.

[48] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[49] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks (IJCNN)*, pages 1453–1460. IEEE, 2011.

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

[51] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 34(11):8135–8153, 2022.