Backdoor Sample Detection through Progressive Perturbation and Loss Trajectory Analysis

YangTan[1], Patrick P. K. Chan[2*], Daniel S. Yeung[3]
[1]Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, Guangzhou, China
Email: buyidao24@163.com
[2]Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, Guangzhou, China
Email: patrickchan@scut.edu.cn
[*]Corresponding author.
[3]Hong Kong
Email: danyeung@ieee.org

***Abstract*—Backdoor attacks compromise deep neural networks (DNNs) by embedding hidden triggers that cause malicious misclassifications, posing serious threats to model reliability. This paper presents a progressive spatial and spectral perturbation (PSSP) framework for detecting poisoned samples through loss trajectory analysis. Each input is sequentially subjected to composite perturbations that combine Gaussian blur and additive Gaussian noise, allowing the model's prediction behavior to be observed under increasing distortion levels. Clean and poisoned samples usually produce different loss trajectories: clean samples tend to show smoother degradation patterns, while poisoned samples often deviate when trigger-dependent cues are weakened. These trajectories are modeled by a lightweight autoencoder trained on a small trusted clean subset from the target task. Detection is performed using reconstruction error under a soft-label black-box setting, where only output probabilities are required and gradients, hidden representations, and model parameters are not accessed. Experiments on CIFAR-10 and GTSRB show that PSSP achieves strong detection performance, with AUROC up to 96.6%, and stable behavior across evaluated datasets, architectures, and attack types. Compared with representative perturbation-based, corruption-based, and recent input-level defenses, PSSP obtains competitive or superior average performance under the evaluated settings. Sensitivity analyses further show that PSSP remains stable under reasonable threshold, perturbation, clean-data, and autoencoder settings, while defense-aware adaptive attack evaluation clarifies its robustness boundary under trajectory-mimicking attacks. Frequency-domain and ablation analyses further indicate that Gaussian blur and additive noise play complementary roles in suppressing trigger-related artifacts, forming an effective spatial-spectral perturbation strategy for practical backdoor sample detection.**

***Index Terms*—Backdoor Attack, Distortion, Sensitivity.**

I. Introduction

Deep neural networks (DNNs) have achieved remarkable success in computer vision tasks such as image classification and object detection [1–3]. However, their growing adoption in security-sensitive applications has exposed them to adversarial threats, particularly backdoor attacks [4–6]. In these attacks, adversaries inject a small fraction of training samples embedded with stealthy trigger patterns. Once trained, the compromised model maintains high accuracy on benign inputs but consistently misclassifies any input containing the trigger. This covert behavior makes backdoor attacks especially dangerous, as standard evaluation on clean data cannot reveal the compromise.

Backdoor research has advanced from early visible triggers (e.g., BadNets [4]) to more sophisticated strategies such as adaptive, invisible, or frequency-domain triggers [7–9]. Recent studies have further explored clean-label sample-specific attacks and multi-trigger multi-target attacks, making the trigger form and attack objective more diverse [10, 11]. These developments expand the threat landscape and make reliable detection increasingly difficult. Consequently, designing defenses that are both robust and generalizable across architectures and attack types has become a pressing challenge.

One major line of defense focuses on detecting and removing poisoned samples from the training set [12–14]. Compared with model-centric approaches [15–19], which require analyzing internal parameters or retraining the model, data-centric methods are more practical in limited-access scenarios, lightweight to deploy, and capable of purifying training data before model release. Recent detection methods have also investigated prediction-consistency cues, such as scaled prediction consistency in SCALE-UP [20] and parameter-oriented scaling consistency in IBD-PSC [21]. Other recent defenses, including MoCC-BD-FID [22] and DACO-BD [23], further indicate that backdoor defense has been explored under different application settings and data modalities. However, existing sample-level detectors often rely on fixed-intensity perturbations, discrete label-changing events, or architecture-specific assumptions, which may limit their robustness when facing diverse or adaptive triggers. Their effectiveness can also degrade when applied across different network architectures [14, 24] or when facing triggers distributed across spatial and frequency domains [7, 9, 25, 26].

Unlike TeCo, which mainly relies on discrete corruption responses and label-changing behavior [14], PSSP models the complete loss trajectory under progressive perturbations. Different from SCALE-UP [20] and IBD-PSC [21], which exploit scaled input prediction consistency or parameter-oriented scaling consistency, PSSP does not use pixel amplification or model-parameter scaling. Instead, it learns the clean loss-

trajectory manifold and identifies poisoned samples through reconstruction errors.

To address these challenges, we propose Progressive Spatial-Spectral Perturbation (PSSP) Based Backdoor Sample Detection, a perturbation-driven framework that identifies poisoned samples by analyzing how model predictions evolve under structured distortions. Instead of relying on a single perturbation level, PSSP applies a short sequence of progressively stronger composite perturbations that combine Gaussian blur and additive Gaussian noise. This process captures the model's behavioral evolution as perturbations intensify, producing a loss trajectory that reflects each sample's prediction stability. Clean samples, which rely on robust semantic representations, exhibit smooth and consistent loss trajectories, whereas poisoned samples, which depend on fragile trigger cues, display abrupt deviations once the trigger information is disrupted. To distinguish normal from abnormal behavior, a lightweight autoencoder is trained on clean loss trajectories extracted from a small trusted clean subset, so that it can learn the manifold of normal prediction patterns. During inference, the reconstruction error serves as a deviation score, where large deviations from the learned manifold indicate potentially poisoned samples. Because PSSP requires only soft-label output probabilities and does not access gradients, hidden features, or model parameters, it operates under a soft-label black-box setting and remains independent of model architecture-specific internal representations. By integrating spatial and spectral perturbation analysis with unsupervised trajectory modeling, PSSP provides a principled, data-efficient, and architecture-agnostic solution for backdoor sample detection, achieving strong empirical performance and balanced detection behavior across diverse attacks and neural network architectures.

The main contributions of this paper are as follows:

- A novel progressive perturbation framework is proposed for backdoor sample detection, where each input is subjected to a sequence of composite Gaussian blur-noise distortions. By analyzing the model's evolving prediction dynamics under increasing perturbation strength, the method transforms each input's response into a distinctive loss trajectory that captures its behavioral sensitivity.
- A lightweight autoencoder-based detector is introduced to learn the manifold of clean loss trajectories in an unsupervised manner, enabling poisoned samples to be identified via reconstruction errors. The approach operates under a soft-label black-box setting, requiring only output probabilities without access to gradients, hidden representations, or model parameters, while using a small trusted clean subset for autoencoder training and threshold calibration.
- Extensive experiments across multiple datasets, architectures, and attack types demonstrate consistent performance and strong cross-architecture robustness. Comparisons with representative and recent input-level defenses further validate its effectiveness under diverse trigger mechanisms. Frequency-domain and ablation analyses show that Gaussian blur and additive noise complement each other in suppressing trigger artifacts, forming a spatial-spectral foundation for lightweight, architecture-agnostic detection. Sensitivity and defense-aware adaptive attack evaluations further clarify the practical stability and robustness boundary of PSSP.

II. Related Works

A. Backdoor Attacks

Backdoor attack techniques have evolved rapidly, progressing from early static trigger designs to more stealthy and adaptive approaches. Traditional static patterns, exemplified by BadNets [4], are increasingly less effective because their conspicuous visual features can be exposed by simple statistical analysis. Modern attacks, by contrast, pursue stealth and adaptability along three major technical directions. *Learnable dynamic triggers* (e.g., LIRA [27]) employ deep networks to automatically generate sample-specific patterns. These triggers adapt to the target model's decision boundary and enhance diversity, making detection harder. *Imperceptible geometric deformations* (e.g., WaNet [8]) apply subtle pixel-level warping or local perturbations, exploiting characteristics of human visual perception to maintain stealth while preserving attack success. On the other hand, *frequency-domain embedding* (e.g., SSBA [25], c-BaN [28]) hides triggers in high-frequency image components, allowing them to evade spatial-domain detection methods.

Moreover, researchers have also explored other dimensions of stealthiness. Color-space perturbations (Color Backdoor) [29] manipulate global color channel distributions, while compression-resistant adaptive frequency triggers are optimized to survive image compression pipelines. Clean-label sample-specific attacks further improve stealthiness by associating triggers with semantic or attribute-level cues, making poisoned samples less visually distinguishable from benign ones [10]. More recently, *multi-trigger attacks* [11, 30–32] have emerged, enabling a single compromised model to respond to multiple trigger types. This breaks the single-target assumption of traditional attacks and significantly increases both the flexibility and potential damage of backdoor threats. These recent developments indicate that backdoor attacks are becoming more diverse in trigger form, target mapping, and deployment scenarios, which increases the difficulty of designing a general detector.

B. Backdoor Defenses

Sample-level detection aims to identify whether individual samples are poisoned and can be used for training-data sanitization or test-time screening [14, 24]. Existing approaches can be broadly divided into two categories. *Input Perturbation Response Methods* exploit the observation that poisoned and clean samples differ in output stability under controlled distortions. Early work such as STRIP [12] and DBR [13] apply fixed-intensity perturbations and measure prediction consistency. More advanced methods, such as TeCo [14], extend this idea by applying multiple perturbation types (e.g., Gaussian blur, JPEG compression, random noise) across several

intensity levels. TeCo then records the critical distortion level at which label flipping occurs, forming a multi-dimensional feature vector. Statistical measures such as entropy or dispersion of this vector are used to discriminate poisoned samples (high dispersion) from clean ones (low dispersion).

Deep Feature Representation Methods analyze hidden representations instead of input perturbations. Techniques such as Activation Clustering and Spectral Signatures [33] examine the geometry of internal feature spaces. By comparing the distribution or spectral properties of clean and poisoned samples, these methods aim to uncover latent anomalies introduced by backdoor triggers. Backdoor defense has also been extended to different application settings and data modalities. MoCC-BD-FID [22] studies multi-objective clustering-combination defense for federated intrusion detection in industrial control systems, while DACO-BD [23] formulates data augmentation selection for SAR image classification as a combinatorial optimization problem. Although these methods differ from the sample-level image classification setting considered in this work, they further highlight the importance of access assumptions, detection signals, and task-specific constraints when comparing backdoor defenses.

Despite these advances, existing sample-level defenses still face several limitations. Perturbation-based methods such as STRIP [12], DBR [13], and TeCo [14] are generally lightweight and easy to deploy, but they often rely on fixed perturbation responses, discrete label-changing events, or handcrafted consistency statistics. As a result, their detection performance may become unstable when triggers are subtle, adaptive, or distributed across spatial and frequency domains. Recent input-level defenses such as SCALE-UP [20] and IBD-PSC [21] further exploit scaled prediction consistency or parameter-oriented scaling consistency. However, SCALE-UP depends on the scaled prediction behavior of inputs, while IBD-PSC requires parameter or BN-layer access, which limits its applicability to architectures without batch-normalization layers. Feature-based methods can reveal hidden representation anomalies, but they usually require access to internal activations and are less suitable for limited-access deployment.

These limitations motivate a detector that avoids model-internal access, reduces reliance on single response statistics, and captures the complete degradation behavior of each sample. In contrast to existing methods, PSSP uses soft-label output probabilities to construct progressive loss trajectories and learns the clean trajectory manifold from a small trusted clean subset. Poisoned samples are then identified according to their reconstruction errors, providing a trajectory-level detection signal under progressive spatial-spectral perturbations.

In summary, the proposed method is closely related to input-level sample detection methods but differs from them in both access assumption and detection representation. Unlike TeCo, which mainly relies on hard-label transition behavior under corruptions, PSSP models the full soft-label loss trajectory. Unlike SCALE-UP, PSSP does not depend on pixel-wise input amplification. Unlike IBD-PSC, PSSP does not require parameter-oriented scaling or batch-normalization-layer access. Compared with feature-based defenses, PSSP also avoids using internal activations or hidden representations. This positioning clarifies that PSSP provides a complementary trajectory-manifold detection signal under a soft-label black-box setting.

III. Proposed Method

The proposed detection framework is motivated by the observation that clean and backdoor-poisoned samples respond differently to progressive distortions. When a model is exposed to increasing levels of Spatial-Spectral Perturbation, the semantic robustness of clean samples allows their predictions to remain relatively stable, while poisoned samples, whose decisions depend on fragile trigger features, exhibit sharp fluctuations in confidence once these features are disrupted. This behavioral divergence can be effectively captured through a sequence of cross-entropy losses, forming what we term a loss trajectory, which reflects how the model's confidence evolves under controlled spatial-spectral perturbations.

To exploit this phenomenon, each input sample is perturbed through a series of gradually intensified distortions that progressively suppress high-frequency trigger information while retaining low- and mid-frequency semantic content. Clean samples therefore yield smooth and gradually changing loss trajectories, whereas poisoned ones show irregular or abrupt changes. A lightweight autoencoder, trained on loss trajectories from a small trusted clean subset, is then used to model the manifold of normal prediction behavior. During inference, reconstruction errors quantify deviations from this learned manifold, providing an unsupervised criterion for identifying poisoned samples. Since the approach requires output probabilities but does not access gradients, hidden features, or model weights, it operates under a soft-label black-box setting. The detector is therefore trained and calibrated for the target task, rather than being assumed to be a universal cross-dataset detector. The framework of the proposed Progressive Spatial-Spectral Perturbation (PSSP) based Backdoor Detection is shown in Figure 1.

As shown in Figure 1, PSSP contains two interconnected flows. The red arrows denote the training and threshold-calibration flow. The potentially poisoned dataset $\mathscr{D}$ is first used to train the classifier f_D, while a small trusted clean subset $\mathscr{D}_c$ is obtained from the target task through random selection and manual verification. Clean samples from $\mathscr{D}_c$ are processed by the same perturbation and loss-trajectory profiling procedure to extract clean loss trajectories, which are then used to train the autoencoder and determine the threshold θ_α.

The blue arrows denote the inference and sample-screening flow. For each candidate sample $x \in \mathscr{D}$, progressive spatial-spectral perturbations generate a sequence of perturbed inputs. The trained classifier f_D produces the corresponding losses under different perturbation levels, and these losses are combined into a loss trajectory $\ell(x)$. The trained autoencoder reconstructs $\ell(x)$ and computes the reconstruction error $\Delta\ell(x)$. Finally, the calibrated threshold is used for decision making. Samples with

$\Delta\ell(x) > \theta_\alpha$ are identified as poisoned, while the remaining samples are regarded as clean and retained in the purified dataset. In the main experiments, $\alpha = 0.05$, corresponding to the 95th-percentile threshold on the trusted clean subset.

A. Problem Setup

Given a training set $\mathscr{D}$ that may contain backdoor-contaminated samples, our objective is to construct a purified subset $\mathscr{D}_p$ by detecting and removing poisoned instances, thereby mitigating the impact of the attack. We consider a soft-label black-box setting, where the defender can query the trained classifier and obtain output probabilities, but cannot access model gradients, hidden-layer representations, training dynamics, or model parameters. We assume access to a small trusted clean subset $\mathscr{D}_c$, which is obtained from the target task through random selection and manual verification, or from other trusted data sources. This subset is used to train the autoencoder and determine the detection threshold. The detector is trained and calibrated for the target dataset and task, rather than being assumed as a universal cross-dataset detector. For each sample $x \in \mathscr{D}$, PSSP computes a loss trajectory under progressive perturbations and assigns an anomaly score according to the reconstruction error. Samples whose reconstruction errors exceed the calibrated threshold are identified as poisoned, while the remaining samples are retained in $\mathscr{D}_p$.

B. Algorithm

a) Progressive Spatial-Spectral Perturbation Generation: To reveal the vulnerability of backdoor triggers, each sample is subjected to a sequence of progressively stronger composite perturbations. The purpose of using multiple distortion levels is to capture the distinct behavioral differences between clean and poisoned samples as perturbation intensity increases. Clean samples, which rely on semantically meaningful and robust features, tend to maintain stable predictions under mild or moderate distortions. In contrast, poisoned samples depend on fragile trigger cues that quickly degrade under stronger perturbations, leading to sharp fluctuations in model confidence.

To systematically expose these behavioral gaps, we employ a spatial-spectral perturbation mechanism that combines Gaussian blur and additive Gaussian noise. Gaussian blur acts as an isotropic low-pass filter that suppresses high-frequency, spatially localized structures commonly used by triggers, such as sharp edges or small textured regions, while preserving mid- and low-frequency semantics crucial for accurate classification. Additive Gaussian noise, which is spectrally white, complements this process by injecting small, uncorrelated disturbances across all frequency bands. It effectively disrupts residual or adaptive trigger alignments that may persist after blurring, especially those encoded in frequency or feature domains, without introducing structured artifacts.

The combined perturbation therefore forms a smooth and continuous spatial-spectral degradation path that weakens both localized spatial triggers and distributed spectral cues in a controlled manner. This dual-domain design provides fine-grained control over distortion strength, making the perturbation process computationally efficient, differentiable, and architecture-independent. The progressive structure of this perturbation series enables the model to reveal how prediction stability changes as perturbation strength increases, offering a reliable foundation for constructing discriminative loss trajectories that distinguish clean and poisoned samples.

Formally, each input sample x, where $(x,y) \in \mathscr{D}$, is transformed into a sequence of progressively perturbed versions:

$$\tilde{x} = \{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_k\}, \tag{1}$$

where each perturbed image $\tilde{x}_i$ is generated as

$$\tilde{x}_i = \mathrm{Blur}_{\sigma_i}(x) + \varepsilon_i. \tag{2}$$

with Blur_{σ_i} denoting Gaussian blurring with standard deviation σ_i, and $\varepsilon_i \sim \mathscr{N}(0, \eta_i^2)$ denotes additive Gaussian noise independently sampled for each pixel and channel. After perturbation, pixel values are clipped to the valid image range.

Spatial Perturbation (Gaussian Blur). The Gaussian blur operator acts as an isotropic low-pass filter that suppresses high-frequency and spatially localized structures, such as trigger edges or patches, while largely preserving mid- and low-frequency semantic content.

The discrete two-dimensional Gaussian kernel is obtained by uniformly sampling its continuous form on a finite $k \times k$ grid with radius $r = \frac{k-1}{2}$. The normalized kernel is defined as

$$\begin{aligned} G_{\sigma_i}(u,v) &= \frac{1}{2\pi\sigma_i^2} \exp\left(-\frac{u^2+v^2}{2\sigma_i^2}\right) \\ &\approx \frac{\exp\left(-\frac{u^2+v^2}{2\sigma_i^2}\right)}{\sum_{a=-r}^{r}\sum_{b=-r}^{r}\exp\left(-\frac{a^2+b^2}{2\sigma_i^2}\right)}, \end{aligned} \tag{3}$$

where (u,v) denotes discrete spatial offsets from the kernel center. The denominator normalizes the kernel so that $\sum_{u,v} G_{\sigma_i}(u,v) = 1$, thereby preserving brightness after convolution.

The blurring operation is implemented through discrete convolution:

$$\mathrm{Blur}_{\sigma_i}(x(p,q)) = \sum_{u=-r}^{r}\sum_{v=-r}^{r} G_{\sigma_i}(u,v)\, x(p-u, q-v), \tag{4}$$

where $x(p,q)$ denotes the spatial coordinates of x. A kernel size of 3×3 ($r = 1$) is used to achieve adequate spatial attenuation without over-smoothing.

Spectral Perturbation (Additive Noise). The additive Gaussian noise component introduces spectrally white perturbations that disrupt subtle feature correlations and residual structured patterns that may persist after spatial blurring. Formally, for each pixel location $x(p,q)$, the additive noise term is sampled as

$$\varepsilon_i(p,q) \sim \mathscr{N}(0, \eta_i^2). \tag{5}$$

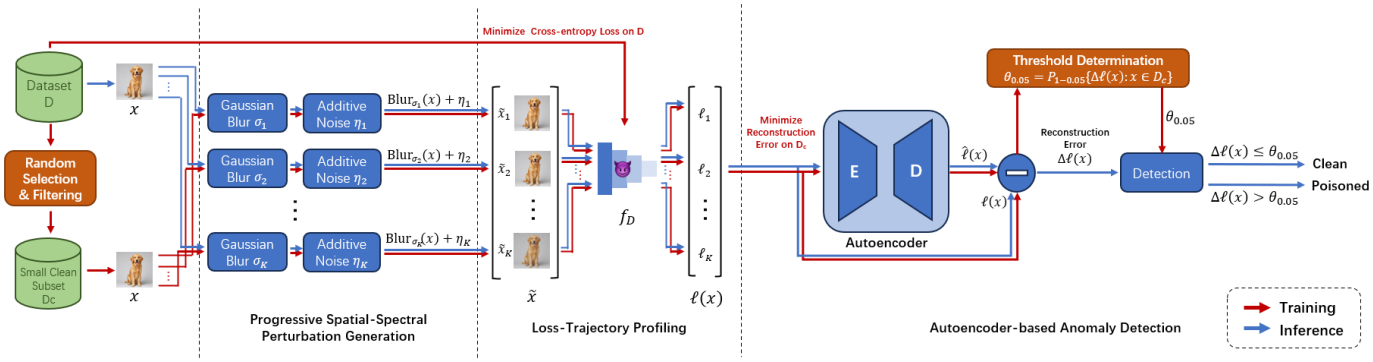

Fig. 1: Framework of the proposed Progressive Spatial-Spectral Perturbation (PSSP) based backdoor sample detection. Red arrows denote the training and threshold-calibration flow based on the trusted clean subset $\mathscr{D}_c$, while blue arrows denote the inference and sample-screening flow for candidate samples from $\mathscr{D}$. The trusted clean subset $\mathscr{D}_c$ is obtained from the target task through random selection and manual verification, and is used for autoencoder training and threshold determination. Candidate samples are identified as poisoned or clean by comparing their reconstruction errors with the calibrated threshold θ_α.

where η_i controls the standard deviation of the zero-mean Gaussian noise applied independently to each pixel and channel. This isotropic perturbation uniformly distributes energy across all frequency bands, weakening the coherence of trigger-dependent activations in both spatial and spectral domains. Unlike structured distortions, additive Gaussian noise does not introduce new artifacts, thereby preserving the semantic integrity of clean samples while effectively degrading fragile trigger cues.

Progressive Perturbation Schedule. To ensure a smooth and monotonic increase in distortion strength, both σ_i and η_i are linearly scaled across perturbation stages:

$$\begin{aligned}\sigma_i &= \sigma_{\min} + \frac{i-1}{\max\{1,k-1\}}(\sigma_{\max}-\sigma_{\min}),\\ \eta_i &= \eta_{\min} + \frac{i-1}{\max\{1,k-1\}}(\eta_{\max}-\eta_{\min}),\end{aligned} \qquad i=1,\ldots,k. \tag{6}$$

We set $\sigma_i \in [0.4, 0.8]$ and $\eta_i \in [0.05, 0.15]$, providing a controlled severity ramp. Early perturbation levels minimally affect benign semantics, while higher levels progressively attenuate trigger-related artifacts. The resulting sequence produces smooth degradation trajectories that effectively differentiate clean and poisoned samples during subsequent analysis.

b) Loss Trajectory Profiling: As perturbations intensify, the prediction stability of each sample can be traced through its loss trajectory. Clean samples generally exhibit smooth and gradual changes in loss, reflecting their reliance on stable semantic features. In contrast, poisoned samples show sharp fluctuations once the perturbation begins to distort or suppress the trigger, revealing their dependence on non-semantic artifacts. This gradual perturbation process therefore exposes the transition from confidence stability to instability, enabling clear separation between clean and contaminated behaviors.

A model $f_{\mathscr{D}}$ is first trained on the dataset $\mathscr{D}$. For each perturbed input $\tilde{x}_i$, the cross-entropy loss L_{CE} with respect to its reference label y is computed as

$$\ell_i = L_{\text{CE}}(f_{\mathscr{D}}(\tilde{x}_i), y), \tag{7}$$

where y is the true label for clean samples and may correspond to the attacker's target for poisoned ones.

The set of losses across perturbation levels forms a *loss trajectory*, which describes how prediction error evolves as distortions increase:

$$\ell(x) = [\ell_1, \ell_2, \ldots, \ell_k]^\top. \tag{8}$$

This vector $\ell(x)$ effectively captures the dynamic sensitivity of the model to perturbations and serves as a discriminative representation for subsequent anomaly detection.

c) Autoencoder-Based Backdoor Attack Detection: Since labeled poisoned data are unavailable, training a supervised classifier to distinguish between clean and contaminated samples is impractical. To overcome this limitation, an unsupervised autoencoder is employed to model the manifold of clean loss trajectories. The core intuition is that clean samples follow consistent and learnable patterns, while poisoned ones deviate from this normal behavior due to their unstable loss dynamics under perturbation. Consequently, the reconstruction error of poisoned samples is expected to be significantly higher than that of clean samples.

Given the low dimensionality of the loss trajectory, a lightweight autoencoder is designed, consisting of an encoder with layers $10 \to 32 \to 16 \to 2$ (each followed by BatchNorm, ReLU, and dropout) and a symmetric decoder that reconstructs trajectories from 2 back to 10 dimensions. The autoencoder is trained on clean loss trajectories from $\mathscr{D}_c$ by minimizing

$$\mathscr{L}_{\text{AE}} = \frac{1}{|\mathscr{D}_c|}\sum_{x\in\mathscr{D}_c}\left\|\ell(x)-\hat{\ell}(x)\right\|_2^2. \tag{9}$$

For each sample x, the reconstruction error is computed as

$$\Delta\ell(x) = \left\|\ell(x)-\hat{\ell}(x)\right\|_2^2. \tag{10}$$

where $\hat{\ell}(x)$ denotes the reconstructed loss trajectory generated by the autoencoder.

During inference, this reconstruction error serves as a deviation score for anomaly detection. Let

$$\mathscr{E}_c = \{\Delta\ell(x) \mid x \in \mathscr{D}_c\} \tag{11}$$

denote the set of reconstruction errors measured on the clean validation set $\mathscr{D}_c$. The threshold is determined by the empirical $(1-\alpha)$-percentile of $\mathscr{E}_c$:

$$\theta_\alpha = \inf\left\{t : \frac{1}{|\mathscr{D}_c|}\sum_{x\in\mathscr{D}_c}\mathbb{I}[\Delta\ell(x) \le t] \ge 1-\alpha\right\}. \tag{12}$$

In the main experiments, α is set to 0.05, so θ_α corresponds to the 95th percentile of reconstruction errors on $\mathscr{D}_c$. The anomaly score is then defined as

$$s(x) = \text{sign}(\Delta\ell(x) - \theta_\alpha). \tag{13}$$

Samples with $s(x) > 0$ are identified as poisoned, while those with $s(x) \le 0$ are classified as clean. This percentile choice provides a robust and model-agnostic balance between false positives and false negatives. Setting θ_α too low leads to excessive rejection of clean samples, while setting it too high allows poisoned ones to pass undetected. Empirically, the 95th percentile consistently yields stable detection across architectures and datasets, supporting the stability of the proposed approach under the evaluated settings.

Finally, the purified dataset $\mathscr{D}_p$ is obtained by retaining samples whose scores are less than or equal to zero from the original dataset $\mathscr{D}$. It is defined as

$$\mathscr{D}_p = \{x \mid x \in \mathscr{D} \text{ and } s(x) \le 0\}. \tag{14}$$

IV. Experiment

A. Experimental Setting

Datasets and Model Architectures: Two widely adopted benchmark datasets in backdoor learning, CIFAR-10 [34] and GTSRB [35], are considered in our experiments. These datasets differ in image complexity and class granularity, allowing the detector to be evaluated under different visual classification settings. In accordance with the standardized evaluation protocol of BackdoorBench [36], two representative neural network architectures are adopted: PreAct-ResNet18 [2] and VGG16 [37]. These models represent residual and traditional convolutional designs, respectively, and are used to examine whether the detection signal remains stable across different backbone structures.

Backdoor Attack Configuration: the backdoor sample detection methods are evaluated against six state-of-the-art backdoor attack methods that cover a wide spectrum of trigger generation strategies: BadNets [4], Blended [6], BPP [7], Input-Aware [26], WaNet [8], and FTrojan [9]. All attacks are implemented with their default hyperparameters as specified in BackdoorBench to ensure reproducibility and fair comparison. The poisoning rate is fixed at 5% of the training set for both CIFAR-10 and GTSRB. Following the standardized BackdoorBench protocol, these attacks are treated as digital backdoor triggers in this work. They cover visible patch, blended, quantization-based, input-adaptive, warping-based, and frequency-domain trigger mechanisms, providing a broad and reproducible benchmark for sample-level detection.

Defense Methods for Comparison: To benchmark the proposed method, we compare it with five representative backdoor sample detection methods. STRIP [12] and DBR [13] are perturbation-based defenses that exploit prediction stability or sample sensitivity under input transformations. TeCo [14] is a strong corruption-based detector that records label-changing behavior under multiple corruption types and intensity levels. SCALE-UP [20] is a recent input-level defense that detects backdoor samples through scaled prediction consistency under pixel-wise amplification. IBD-PSC [21] further examines parameter-oriented scaling consistency and is included as a recent representative defense with stronger access assumptions. In our evaluated backbone settings, IBD-PSC is applied to PreAct-ResNet18 but marked as not applicable to the adopted VGG16 backbone, because IBD-PSC relies on BN-layer parameters for parameter-oriented scaling. The adopted VGG16 backbone does not contain BN layers, and adding extra BN layers would change the evaluated model architecture. Therefore, IBD-PSC is not applied under the VGG16 setting. For fair comparison, all defenses are evaluated on the same poisoned training sets and attack configurations. PSSP operates under a soft-label black-box setting and uses a small trusted clean subset $\mathscr{D}_c$ to train the autoencoder and calibrate the detection threshold. In our experiments, $\mathscr{D}_c$ comprises 2% of the training data and follows the same distribution as the full training dataset $\mathscr{D}$. For baselines that require or benefit from a clean split, the same clean data budget is provided. For methods that do not require a trusted clean subset, we follow their original evaluation assumptions.

Evaluation Metrics: Detection performance is evaluated by using three complementary metrics: *the True Positive Rate (TPR)*, which measures the proportion of clean samples correctly identified as benign; *the True Negative Rate (TNR)*, which reflects the proportion of poisoned samples successfully detected; and *the Area Under the ROC Curve (AUROC)*, which quantifies the overall separability between clean and poisoned samples by evaluating the trade-off between true positive and false positive rates across varying thresholds. Using both AUROC and the fixed-threshold TPR/TNR results allows us to evaluate not only ranking-based separability, but also the practical balance between retaining clean data and filtering poisoned data.

B. Experimental Results and Analysis

1) Detection Performance: **Overall Detection Performance:** The quantitative results in Tables I and II show that the proposed method achieves strong and stable detection performance across the evaluated datasets and architectures. On CIFAR-10 with PreAct-ResNet18, PSSP obtains an average AUROC of 96.57%, outperforming TeCo [14] (89.09%), SCALE-UP [20] (66.98%), IBD-PSC [21] (82.26%), STRIP [12], and DBR [13]. On GTSRB

with PreAct-ResNet18, PSSP also obtains the highest average AUROC among the applicable baselines, reaching 93.53%. These results indicate that modeling the full soft-label loss trajectory provides a more informative detection signal than relying only on fixed perturbation responses, label-changing events, or scaling-consistency cues.

In addition to average AUROC, the STD columns provide evidence about cross-attack stability. PSSP obtains relatively low AUROC standard deviations across the evaluated attack types, suggesting that its detection signal is less dependent on a single trigger mechanism in the tested settings. Table II further reports the fixed-threshold behavior. With the 95th-percentile threshold calibrated on the trusted clean subset, PSSP achieves an average TPR/TNR of 95.28%/93.19% on CIFAR-10 with PreAct-ResNet18. This result shows a balanced trade-off between retaining clean samples and filtering poisoned samples, which is important because overly aggressive filtering may damage the integrity of the purified training set.

Robustness to Trigger Type: The evaluated attacks cover visible patch, blended, input-adaptive, warping-based, quantization-based, and frequency-domain triggers. Conventional approaches, such as STRIP [12] and DBR [13], mainly depend on fixed-intensity perturbations to measure prediction instability. SCALE-UP exploits scaled prediction consistency under pixel-wise amplification, while IBD-PSC relies on parameter-oriented scaling consistency. These cues are useful, but their effectiveness can vary when the trigger behavior does not match the assumed consistency pattern or when the required model structure is unavailable.

Across the six evaluated attack types, PSSP maintains AUROC values above 82% in all tested settings and achieves particularly high performance on Input-Aware, WaNet, BPP, and FTrojan. This result suggests that progressive spatial-spectral perturbations can expose abnormal prediction behavior for different trigger mechanisms. Clean samples usually exhibit smoother and more gradual confidence changes, while poisoned samples tend to show sharper changes in loss trajectories once their trigger-related cues are weakened. By learning this loss-trajectory pattern with an autoencoder trained on trusted clean samples, PSSP shows stable performance across the evaluated trigger types. This finding supports the effectiveness of the proposed detection principle, while the evaluation of additional emerging triggers remains an important direction for future work.

Cross-Architecture Generalization: We further examine whether the detection performance remains stable when the backbone architecture changes from PreAct-ResNet18 to VGG16. As shown in Table I, TeCo shows a clear performance decrease under this transition, especially on CIFAR-10. SCALE-UP also decreases from 66.98% to 50.43% on CIFAR-10 and remains relatively low on GTSRB. IBD-PSC is evaluated on PreAct-ResNet18 but is not applicable to the adopted VGG16 backbone because this VGG16 implementation does not contain the batch-normalization layers required for parameter-oriented scaling. These observations indicate that some baseline methods can be affected by specific perturbation responses, prediction-consistency assumptions, or model-structure requirements.

In comparison, PSSP shows a smaller average AUROC decrease when moving from PreAct-ResNet18 to VGG16, from 96.57% to 93.49% on CIFAR-10 and from 93.53% to 89.09% on GTSRB. Since PSSP relies only on observable output probabilities and does not access gradients, hidden representations, or model parameters, its detection signal is less tied to a particular internal network structure. The trajectory-level representation also reduces the dependence on a single perturbation response. These results support the architecture-agnostic design goal of PSSP in the evaluated backbone settings, although further validation on more architectures would be useful in future work.

2) Reconstruction-Error Distributions After AE: To assess the discriminative effectiveness of the lightweight autoencoder (AE) detector, the reconstruction error distributions of clean and poisoned samples are analyzed under six representative attacks, including BadNet, Blended, Input-Aware, WaNet, BPP, and FTrojan. Experiments are conducted on the CIFAR-10 dataset using the PreAct-ResNet18 architecture with a poisoning rate of 5% and ten progressive perturbation stages. Each sample's loss trajectory, consisting of cross-entropy losses across all perturbation levels, is processed by an AE trained solely on clean trajectories. The squared reconstruction error serves as the anomaly score.

Figure 2 presents the boxplots of reconstruction errors for the six attack types. Across all cases, poisoned samples generally exhibit higher reconstruction errors than clean ones, indicating that their loss trajectories deviate from the clean trajectory manifold after AE reconstruction. For BadNet and Blended, the separation remains particularly evident, which is consistent with the observations discussed above. A similar trend is also observed for Input-Aware, WaNet, BPP, and FTrojan, although the degree of overlap varies across attacks. This result suggests that the proposed loss-trajectory representation is not limited to a single visible-trigger case, but can provide useful anomaly evidence for dynamic, warping-based, quantization-based, and frequency-domain backdoor patterns in the evaluated benchmark. Overall, these results provide direct empirical support for the proposed AE-based anomaly detection mechanism.

3) Frequency-Domain Effects of Composite Gaussian Blur-Noise Perturbation on Backdoor Triggers: To further clarify how the proposed composite perturbation suppresses trigger-related artifacts while preserving semantic structures, we analyze its effects in the frequency domain from both qualitative and quantitative perspectives.

Frequency-Domain Visualization. As a representative qualitative example, Figure 3 shows how progressive spatial-spectral perturbations affect clean, BadNet, and Blended samples. Each column represents a sample type, and each row corresponds to increasing perturbation strength, with residual maps highlighting high-frequency content. At the original level, clean samples mainly show natural frequency patterns around object edges, while BadNet and Blended samples

TABLE I: AUROC (%) comparison of different backdoor sample detection methods across datasets, backbone architectures, and attack types. AVG and STD report the mean and standard deviation across the six attacks, respectively. Bold values indicate the best AUROC or AVG and the lowest STD within each dataset-backbone block. "–" indicates that IBD-PSC is not applicable to VGG16 because the adopted VGG16 backbone does not contain the batch-normalization layers required for parameter-oriented scaling.

Dataset	Backbone	Method	BadNet	Blended	Input-Aware	WaNet	BPP	FTrojan	AVG	STD
CIFAR-10	PreAct-ResNet18	STRIP	75.98	72.64	28.36	39.54	89.21	78.52	64.04	24.22
		DBR	80.01	76.81	56.85	54.84	62.82	78.10	68.24	11.38
		SCALE-UP	61.67	72.20	63.83	50.04	71.91	82.21	66.98	11.03
		IBD-PSC	**97.20**	**95.77**	57.63	81.05	67.93	93.97	82.26	16.47
		TeCo	94.43	94.24	91.59	69.91	89.65	94.70	89.09	9.60
		Ours	93.19	92.21	**99.39**	**99.11**	**99.22**	**96.27**	**96.57**	**3.22**
CIFAR-10	VGG16	STRIP	72.17	71.95	35.74	41.39	84.12	71.43	62.80	19.45
		DBR	78.16	72.31	57.46	48.52	53.78	85.93	66.03	14.92
		SCALE-UP	48.82	48.21	41.06	43.86	71.58	49.07	50.43	10.84
		IBD-PSC	–	–	–	–	–	–	–	–
		TeCo	91.94	**92.43**	60.70	82.92	40.13	90.38	76.42	21.44
		Ours	**95.25**	83.59	**99.95**	**91.25**	**96.16**	**94.76**	**93.49**	**5.60**
GTSRB	PreAct-ResNet18	STRIP	67.24	73.12	26.48	27.63	62.13	57.63	52.37	20.28
		DBR	46.35	54.26	59.78	54.70	62.82	57.70	55.94	5.68
		SCALE-UP	41.78	30.30	29.61	48.04	60.37	63.08	45.53	14.38
		IBD-PSC	**99.02**	70.95	61.33	77.48	64.34	99.73	78.81	16.88
		TeCo	77.61	72.98	83.49	98.79	**94.63**	96.31	87.30	10.77
		Ours	89.57	**88.25**	**91.83**	**99.95**	91.57	**100.00**	**93.53**	**5.16**
GTSRB	VGG16	STRIP	67.13	76.37	32.78	51.03	58.94	47.41	55.61	15.39
		DBR	53.78	54.65	50.35	52.98	51.65	67.94	55.23	6.41
		SCALE-UP	40.39	41.49	26.32	45.47	48.05	68.16	44.98	13.63
		IBD-PSC	–	–	–	–	–	–	–	–
		TeCo	79.00	**90.49**	67.34	43.04	69.15	79.43	71.41	16.20
		Ours	**91.66**	90.12	**86.24**	**93.84**	**82.83**	**89.86**	**89.09**	**3.95**

TABLE II: TPR/TNR (%) comparison of different backdoor sample detection methods across attack types. Clean samples are treated as positive samples, and each cell reports *TPR/TNR*. AVG and STD report the mean and standard deviation across the six attacks, respectively. Bold values indicate the best TPR/TNR or the lowest STD within each dataset-backbone block. "–" indicates that IBD-PSC is not applicable to VGG16 because the adopted VGG16 backbone does not contain the batch-normalization layers required for parameter-oriented scaling.

Dataset	Backbone	Method	BadNet	Blended	Input-Aware	WaNet	BPP	FTrojan	AVG	STD
CIFAR-10	PreAct-ResNet18	STRIP	77.89/65.36	68.67/71.07	90.95/10.90	70.87/32.62	77.13/89.44	65.84/68.90	75.22/56.38	9.03/28.91
		DBR	69.60/79.07	67.36/75.10	38.75/65.02	40.85/69.17	33.04/75.83	71.75/72.02	53.56/72.70	17.78/**5.06**
		SCALE-UP	63.59/58.68	63.87/78.72	74.11/52.88	57.80/43.68	89.85/54.78	74.99/86.04	70.70/62.46	11.49/16.36
		IBD-PSC	94.58/**99.80**	91.54/**99.98**	52.71/62.78	68.07/78.69	92.57/52.18	89.59/**99.76**	81.51/82.20	17.14/21.09
		TeCo	**99.58**/93.56	**95.35**/89.68	**99.46**/88.29	77.71/70.02	**99.98**/87.92	**99.46**/88.29	95.26/86.29	8.77/8.24
		Ours	90.22/89.09	85.19/86.33	98.13/**92.48**	**100.00/99.77**	**99.98/99.00**	98.13/92.48	**95.28/93.19**	**6.13**/5.33
CIFAR-10	VGG16	STRIP	67.89/75.24	60.47/79.07	47.85/59.42	45.17/69.14	78.27/79.43	57.32/82.19	59.50/74.08	12.41/**8.48**
		DBR	10.09/**100.00**	14.27/**96.80**	41.33/62.00	**95.27**/5.91	85.60/16.03	76.57/87.64	53.86/61.40	37.09/41.40
		SCALE-UP	1.83/97.37	13.28/85.27	36.82/61.67	0.97/**99.63**	89.84/53.99	40.74/59.40	30.58/76.22	33.63/20.33
		IBD-PSC	-/-	-/-	-/-	-/-	-/-	-/-	-/-	-/-
		TeCo	**97.10**/71.06	**99.95**/80.02	61.71/64.41	80.58/68.03	38.98/52.03	87.13/**91.24**	77.58/71.13	23.34/13.44
		Ours	93.77/92.36	81.12/70.12	**100.00/97.87**	83.57/90.49	**100.00/96.82**	**92.67**/90.18	**91.86/89.64**	**8.01**/10.09
GTSRB	PreAct-ResNet18	STRIP	58.67/67.54	58.47/78.79	7.57/76.84	9.93/78.23	67.83/54.32	51.95/63.23	42.40/69.83	26.56/**9.89**
		DBR	5.31/97.54	25.73/88.19	32.31/82.80	47.22/60.71	63.18/57.31	71.04/51.38	40.80/72.99	24.56/18.94
		SCALE-UP	3.23/91.78	15.78/79.23	12.93/67.83	6.64/81.92	89.66/50.38	54.13/67.60	30.40/73.12	34.34/14.41
		IBD-PSC	**99.46/99.13**	63.22/**88.93**	77.76/42.33	98.77/50.57	**99.47**/50.33	99.46/99.99	89.69/71.88	15.57/26.89
		TeCo	94.18/51.46	88.10/54.59	**78.06**/88.03	98.67/95.89	94.54/**92.21**	97.38/95.17	**91.82**/79.56	**7.67**/20.76
		Ours	61.87/95.42	**94.69**/55.85	75.73/**95.29**	**98.80/99.13**	87.65/85.75	**100.00/100.00**	86.46/**88.57**	15.00/16.81
GTSRB	VGG16	STRIP	73.72/65.67	60.47/67.64	10.28/69.83	54.19/73.38	57.12/72.17	53.37/43.28	51.53/65.33	21.52/**11.17**
		DBR	23.80/83.59	25.73/**88.19**	4.90/**97.51**	55.66/51.05	43.99/61.57	48.82/83.23	33.82/77.52	19.01/17.54
		SCALE-UP	1.37/**97.43**	15.76/86.68	3.53/95.34	9.79/79.37	92.73/18.61	60.12/68.88	30.55/74.39	37.34/29.28
		IBD-PSC	-/-	-/-	-/-	-/-	-/-	-/-	-/-	-/-
		TeCo	**88.57**/52.06	91.14/75.98	47.30/79.06	42.33/45.27	58.12/**74.91**	67.32/78.93	65.80/67.70	20.57/14.99
		Ours	78.40/93.66	**98.63**/74.31	**72.21**/95.90	**83.21/92.62**	**99.85**/64.28	**87.47/85.94**	**86.63/84.45**	**11.02**/12.61

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2026.3707425

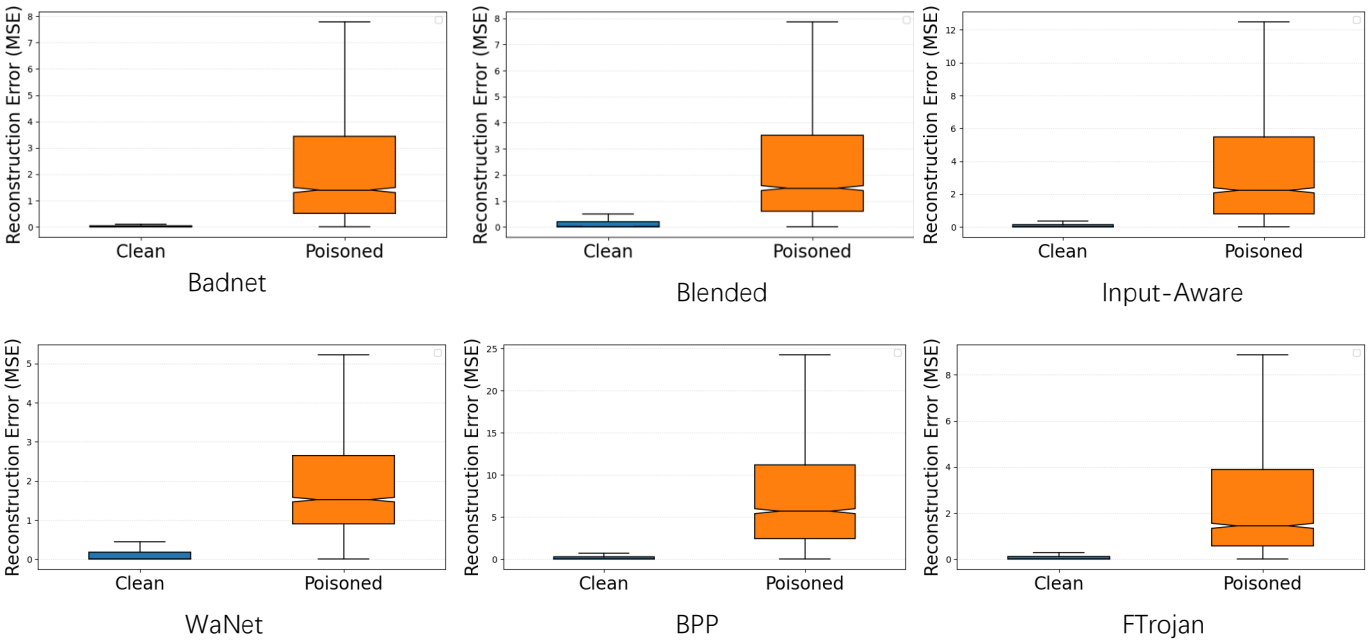

Fig. 2: Reconstruction-error distributions after autoencoder reconstruction under six attack settings on CIFAR-10 with PreAct-ResNet18. For each attack, clean and poisoned samples are evaluated using the corresponding attack-specific classifier and autoencoder. Poisoned samples generally show larger reconstruction errors than clean samples, indicating that their loss trajectories deviate from the learned clean trajectory manifold.

contain additional trigger-related residual structures. BadNet triggers create sharp and localized residual responses, whereas Blended triggers introduce smoother and more distributed frequency components. As perturbations intensify, Gaussian blur attenuates sharp high-frequency structures, and additive Gaussian noise further weakens residual trigger patterns. The visualization supports the intuition that the proposed perturbation can suppress trigger-related artifacts while retaining the main semantic structure of clean samples.

4) Quantitative High-Frequency Energy Analysis: To characterize how perturbations suppress fine structural details, the high-frequency energy $E_H(x)$ of an image x is computed from its Fourier magnitude spectrum. After a spatial Fourier transform is applied, the outer 75% of the spectrum (corresponding to high-frequency regions) is isolated using a circular mask, and the squared magnitudes of these components are summed:

$$E_H(x) = \sum_{u,v} \mathscr{M}(u,v) \left| \mathscr{F}\{x\}(u,v) \right|^2, \tag{15}$$

where $\mathscr{M}(u,v)$ equals 1 for high-frequency coordinates and 0 otherwise. The value of $E_H(x)$ therefore represents the total spectral energy carried by edges, textures, and other fine-scale image structures.

For each perturbation stage i, the change in high-frequency energy and the corresponding loss increment are defined as

$$\Delta E(i) = E_H(x) - E_H(\tilde{x}_i), \tag{16}$$

$$\Delta L(i) = L_{\mathrm{CE}}\big(f_{\mathscr{D}}(\tilde{x}_i), y\big) - L_{\mathrm{CE}}\big(f_{\mathscr{D}}(x), y\big), \tag{17}$$

where $f_{\mathscr{D}}$ denotes the trained classifier and L_{CE} the cross-entropy loss. In this context, $\Delta E(i)$ measures the extent of fine-scale energy suppression, while $\Delta L(i)$ quantifies the model's sensitivity to such spectral degradation.

We extend this quantitative analysis to all six evaluated attacks, including BadNet, Blended, Input-Aware, WaNet, BPP, and FTrojan. As shown in Figure 4, the high-frequency energy reduction generally increases as the perturbation level becomes stronger, confirming that the proposed spatial-spectral perturbation progressively suppresses fine-scale spectral details. The exact reduction pattern varies across attacks, reflecting different trigger characteristics and spectral distributions.

The corresponding loss-increment trajectories are shown in Figure 5. Poisoned samples generally exhibit faster and larger loss increases than clean samples under progressive perturbations. This indicates that trigger-dependent predictions are more sensitive to the proposed perturbation process, since the cues supporting the backdoor behavior are gradually weakened. Although the separation strength differs across attack types, the trend is consistently observed for visible patch, blended, dynamic, warping-based, quantization-based, and frequency-domain triggers.

These results show that high-frequency suppression and prediction instability jointly support the proposed loss-trajectory detection principle. The perturbation process does not rely on a specific trigger form. Instead, it exposes how different poisoned samples deviate from clean samples through their characteristic degradation trajectories.

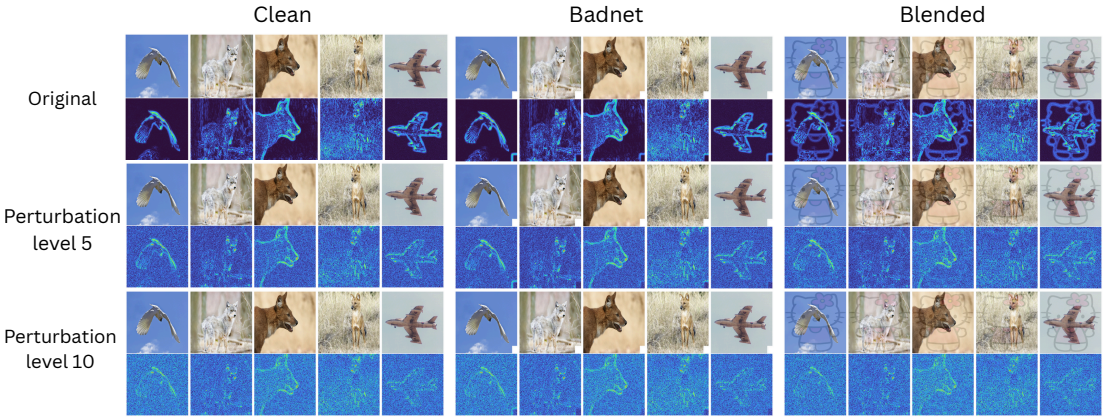

Fig. 3: Visualization of clean, BadNet, and Blended samples under increasing spatial-spectral perturbation levels. Residual maps highlight high-frequency content and illustrate how trigger-related structures are gradually weakened by the composite perturbation.

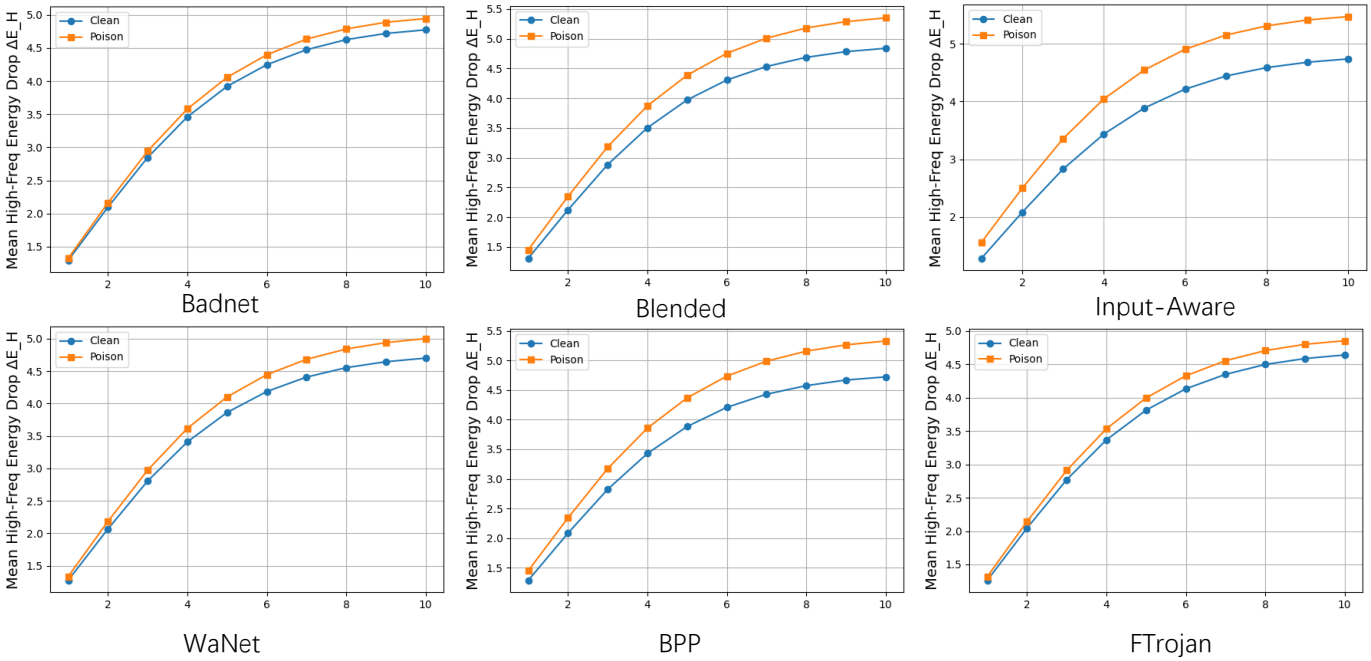

Fig. 4: High-frequency energy reduction curves under progressive perturbations for six attacks on CIFAR-10 with PreAct-ResNet18. The curves show that the proposed perturbation gradually suppresses spectral details, providing a frequency-domain explanation for the loss-trajectory separation.

5) Investigation on Perturbation Methods: To disentangle the individual and joint effects of *Spatial Perturbation* (Gaussian blur) and *Spectral Perturbation* (additive Gaussian noise), we conduct a controlled ablation study. Experiments are performed on CIFAR-10 with PreAct-ResNet18 under a 5% poisoning rate, following the BackdoorBench protocol. Three configurations are compared: (1) **Spatial Perturbation**, implemented using a 3×3 Gaussian kernel with progressively increasing standard deviation σ_k (0.4–0.8); (2) **Spectral Perturbation**, achieved by injecting zero-mean Gaussian noise with standard deviation η_k (0.05–0.15) while omitting blurring; and (3) **Spatial-Spectral Perturbation**, which combines the two perturbations using the same parameter scales.

As shown in Table III, Spatial Perturbation alone already achieves strong AUROC values above 85% for all attacks, indicating that Gaussian blur is effective in weakening localized high-frequency trigger cues. Spectral Perturbation alone gives lower average performance, but it performs well on Input-Aware, suggesting that additive noise can help disrupt certain non-local or adaptive trigger patterns. The combined Spatial-Spectral Perturbation achieves the best performance for all six attacks, with AUROC values ranging from 92.21% to

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2026.3707425

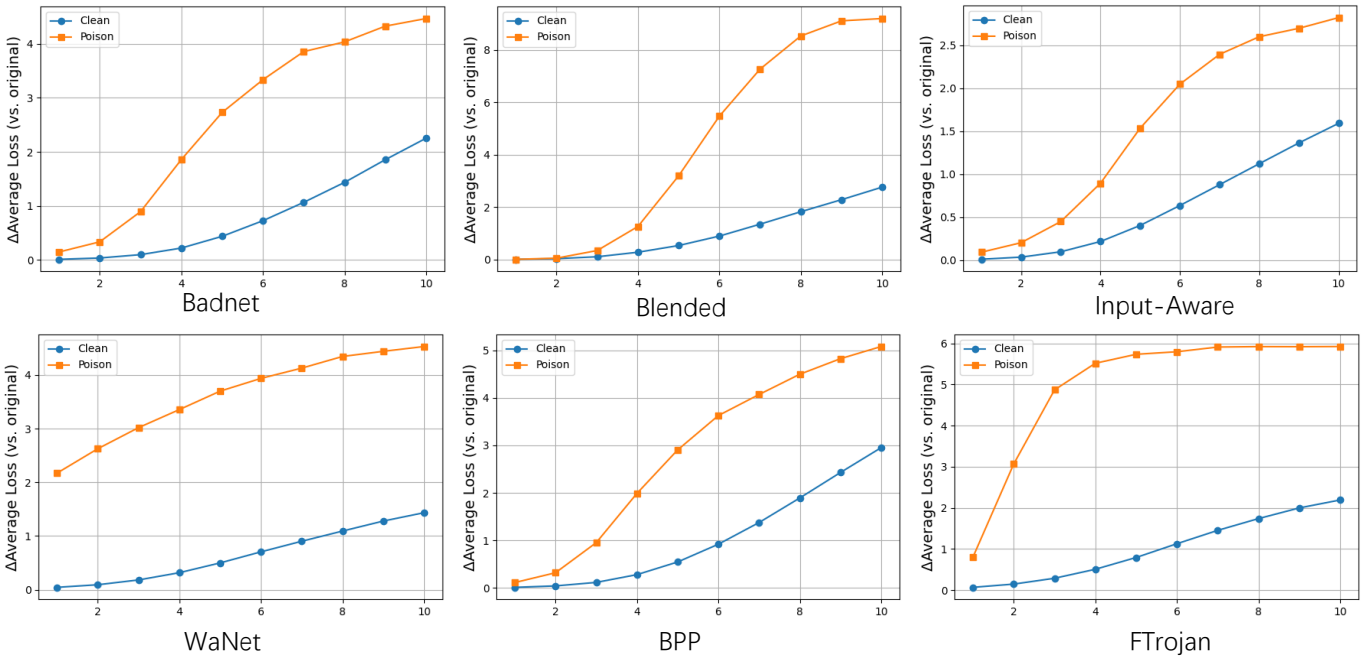

Fig. 5: Average loss-increment trajectories under progressive perturbations for six attacks on CIFAR-10 with PreAct-ResNet18. Poisoned samples generally exhibit faster and larger loss increases, showing that trigger-dependent predictions are more sensitive to the proposed perturbation process.

99.39%. These results support the complementary roles of the two perturbations: spatial blurring suppresses localized high-frequency structures, while spectral noise further destabilizes residual trigger-related patterns.

This ablation also shows that the improvement of PSSP does not come from an arbitrary perturbation choice. Instead, the two components provide different but complementary degradation effects. Spatial Perturbation contributes the main suppression effect in most attacks, while Spectral Perturbation improves the separability for attacks whose trigger behavior is less localized. The combined setting is therefore used as the default configuration in the main experiments.

TABLE III: Ablation comparison of AUROC (%) under different perturbation configurations on CIFAR-10 with PreAct-ResNet18. Spatial Perturbation denotes Gaussian blur, Spectral Perturbation denotes additive Gaussian noise, and Spatial-Spectral Perturbation denotes their combined use in PSSP.

Attack	Spatial Perturbation	Spectral Perturbation	Spatial-Spectral Perturbation
BadNet	92.27	65.76	**93.19**
Blended	90.43	60.56	**92.21**
BPP	91.34	58.54	**99.22**
FTrojan	93.57	62.46	**96.27**
Input-Aware	85.76	88.41	**99.39**
WaNet	89.46	55.98	**99.11**

6) Effect of Perturbation Stages on Detection Performance: The influence of the number of progressive perturbation stages (n) on detection accuracy is analyzed under the CIFAR-10 / PreAct-ResNet18 setting with a 5% poisoning rate. For each n, an autoencoder is trained on clean loss trajectories of length n, while the perturbation schedule follows the same linear interpolation of blur and noise strengths defined previously. The classifier remains fixed, and only the perturbation granularity varies. The AUROC trend with respect to n is shown in Figure 6.

The results show that AUROC generally improves as n increases and then tends to stabilize after sufficient perturbation diversity is available. Attacks such as *BadNet*, *BPP*, *WaNet*, *Input-Aware*, and *FTrojan* already reach high performance around $n = 4$. In contrast, *Blended*, which uses a more distributed low-frequency trigger, benefits from a longer schedule and reaches stronger performance when n approaches 10. This indicates that different trigger types may require different trajectory resolutions to expose their degradation behavior.

Although $n = 4$ provides a good efficiency-accuracy trade-off for several attacks, we use $n = 10$ in the main experiments to maintain a unified setting and to better handle low-frequency or distributed triggers such as *Blended*. This choice is also consistent with the autoencoder architecture described in Section III, where the default loss trajectory has length 10.

7) Sensitivity to Perturbation Schedule: To further examine the influence of perturbation strength, we evaluate PSSP under three perturbation schedules: low, original, and high. The original schedule is the default setting used in the main experiments, where σ gradually increases from 0.4 to 0.8 and η gradually increases from 0.05 to 0.15. The low schedule reduces both ranges by a factor of 0.5, resulting in blur strength

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2026.3707425

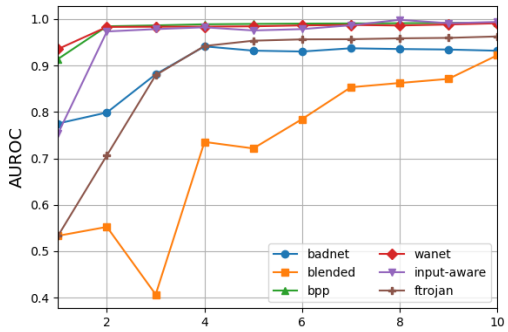

Fig. 6: Effect of the number of progressive perturbation stages (n) on AUROC across different backdoor attack types on CIFAR-10 with PreAct-ResNet18. Larger n provides a finer loss trajectory, while the main experiments use $n = 10$ for a unified setting across attacks.

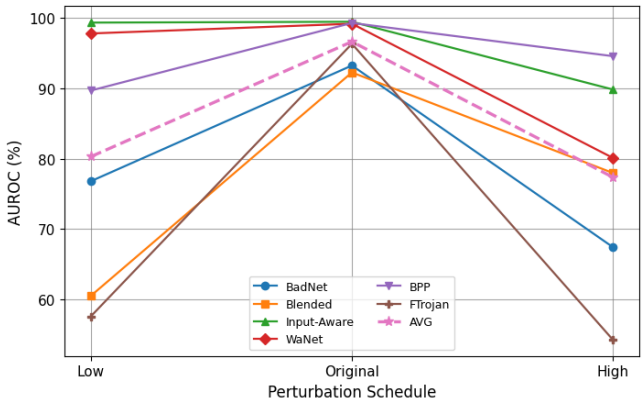

Fig. 7: Sensitivity of PSSP to different perturbation schedules on CIFAR-10 with PreAct-ResNet18. The Low and High schedules are obtained by scaling the default blur and noise ranges by 0.5 and 1.5, respectively. Each curve reports AUROC under one attack type, and AVG denotes the average AUROC across attacks.

from 0.2 to 0.4 and noise level from 0.025 to 0.075. The high schedule increases both ranges by a factor of 1.5, resulting in blur strength from 0.6 to 1.2 and noise level from 0.075 to 0.225. This setting preserves the same progressive increasing pattern and isolates the influence of perturbation strength. Experiments are conducted on CIFAR-10 with PreAct-ResNet18, while the autoencoder architecture, clean data budget, and threshold parameter are kept unchanged. The results are shown in Figure 7.

The original schedule achieves the best average AUROC of 96.57% and the most stable performance across attacks. When the perturbation range is reduced, the average AUROC drops to 80.27%, indicating that weak perturbations are insufficient to suppress trigger-related cues and cannot produce clearly separable loss trajectories. When the perturbation range is increased, the average AUROC decreases to 77.34%. This suggests that overly strong perturbations may also damage semantic information in clean samples and introduce excessive fluctuations into clean trajectories, thereby reducing the reliability of reconstruction-error-based separation. These results show that the perturbation schedule should balance trigger suppression and semantic preservation. The original schedule provides such a balance and is therefore adopted as the default setting.

8) Sensitivity to Threshold Calibration: To examine the influence of threshold calibration, we evaluate PSSP under different values of α. According to the threshold definition in Section III, θ_α is determined by the empirical $(1-\alpha)$-percentile of reconstruction errors on the trusted clean subset. In this work, clean samples are treated as positive samples, while poisoned samples are treated as negative samples. Therefore, TPR measures the proportion of clean samples correctly retained, and TNR measures the proportion of poisoned samples correctly filtered. Experiments are conducted on CIFAR-10 with PreAct-ResNet18, while the attack configuration, perturbation schedule, and autoencoder architecture are kept unchanged. The results are reported in Table IV.

As shown in Table IV, α controls the trade-off between clean-sample retention and poisoned-sample filtering. When $\alpha = 0.01$, the threshold is determined by the 99th percentile of clean reconstruction errors. This setting retains most clean samples and achieves a high average TPR of 98.40%, but the average TNR decreases to 56.34%, indicating that many poisoned samples can still pass the detector. As α increases, the percentile used for threshold calibration decreases, and the detector becomes more aggressive in filtering suspicious samples. Consequently, the average TPR decreases from 98.40% to 84.16%, while the average TNR increases from 56.34% to 97.30%. The default setting $\alpha = 0.05$, corresponding to the 95th percentile, achieves a balanced average TPR/TNR of 95.28%/93.19%. This result supports the use of $\alpha = 0.05$ in the main experiments, as it preserves most clean samples while maintaining effective poisoned-sample rejection.

9) Sensitivity to Clean Data Budget: We further evaluate the influence of the trusted clean subset size on PSSP. The clean subset $\mathcal{D}_c$ is used to train the autoencoder and calibrate the threshold θ_α, so its size may affect both the learned clean trajectory manifold and the stability of threshold estimation. To examine this factor, we reduce the clean data budget from the default 2.00% to 1.00% and 0.50% of the training set. Experiments are conducted on CIFAR-10 with PreAct-ResNet18, while the attack configuration, perturbation schedule, autoencoder architecture, and threshold parameter $\alpha = 0.05$ are kept unchanged. The detection performance is shown in Figure 8.

The results show that PSSP is relatively insensitive to the clean data budget. This is mainly because the autoencoder models the low-dimensional loss-trajectory pattern rather than the full image distribution. Under the fixed progressive perturbation process, clean samples usually exhibit stable trajectory responses, so a small trusted clean subset is sufficient to

TABLE IV: Sensitivity of PSSP to the threshold parameter α on CIFAR-10 with PreAct-ResNet18. Clean samples are treated as positive samples, and each cell reports TPR/TNR (%).

α	BadNet	Blended	Input-Aware	WaNet	BPP	FTrojan	AVG	STD
0.01	98.37/40.25	93.48/63.28	99.38/73.28	100.00/87.34	100.00/53.92	99.17/19.96	98.40/56.34	2.49/24.02
0.05	90.22/89.09	85.19/86.33	98.13/92.48	100.00/99.77	99.98/99.00	98.13/92.48	95.28/93.19	6.13/5.33
0.10	87.37/92.48	80.02/89.37	89.38/93.28	93.48/99.84	93.07/99.15	90.73/95.16	89.01/94.88	4.96/4.04
0.15	82.55/97.37	77.31/91.43	86.65/96.25	87.88/99.98	88.17/99.97	82.40/98.77	84.16/97.30	4.21/3.23

capture the main clean trajectory manifold and calibrate the threshold. Once the main clean trajectory distribution has been captured, increasing the clean subset size brings only marginal improvement. This explains why the average AUROC remains unchanged when the clean budget is reduced from 2.00% to 1.00%, and only decreases slightly from 96.57% to 96.15% when the budget is further reduced to 0.50%.

Across individual attacks, stable performance is also observed. For most attacks, including Blended, WaNet, BPP, and FTrojan, the AUROC remains close to or even higher than that obtained with the 2.00% clean budget. A relatively larger decrease is observed for BadNet, where the AUROC drops from 93.19% to 90.01% at the 0.50% budget. This decrease is mainly related to the limited representativeness of the trusted clean subset under the very small clean budget. Since PSSP trains the autoencoder only on clean loss trajectories, a very small clean subset may not sufficiently cover the diversity of normal local textures, edges, and background patterns. As a result, the learned clean trajectory manifold becomes less complete, and the percentile-based threshold calibration may also become less stable. This effect is more visible for BadNet because its AUROC under the default 2.00% budget is already lower than those of several other attacks, indicating a smaller separation margin between clean and poisoned trajectories. When the clean manifold is estimated from fewer samples, this smaller margin is more easily affected, leading to a relatively larger AUROC decrease. Nevertheless, the overall performance remains high, suggesting that the clean loss-trajectory manifold can still be effectively learned from a very small trusted clean subset.

10) Sensitivity to the Autoencoder Bottleneck Dimension: We further examine the effect of the autoencoder bottleneck dimension on CIFAR-10 with PreAct-ResNet18. The bottleneck dimension controls the compression level of the loss trajectory in the latent space. A very small bottleneck may lose useful trajectory information, while an overly large bottleneck may reconstruct abnormal trajectories too well and weaken the anomaly signal. Therefore, we evaluate bottleneck dimensions of 2, 4, and 8 while keeping the perturbation schedule, clean data budget, and threshold setting unchanged.

As shown in Figure 9, PSSP is relatively insensitive to the bottleneck dimension. When the bottleneck dimension increases from 2 to 8, the average AUROC changes only slightly from 96.57% to 96.74%. This small variation indicates that the discriminative information in the loss trajectory can already be captured in a low-dimensional latent space. Although the

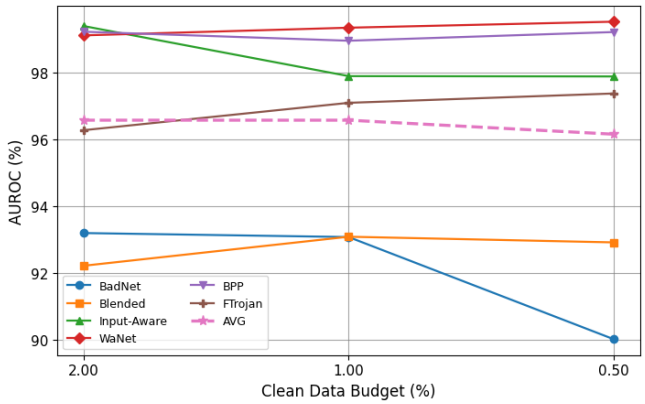

Fig. 8: Sensitivity of PSSP to the clean data budget on CIFAR-10 with PreAct-ResNet18. The clean budget denotes the proportion of training data used as the trusted clean subset $\mathscr{D}_c$. Each curve reports AUROC under one attack type, and AVG denotes the average AUROC across attacks.

bottleneck dimension of 8 gives the highest average AUROC, the improvement over dimension 2 is only 0.17%. In addition, dimension 2 achieves the lowest standard deviation across attacks, indicating more stable performance. Therefore, we use bottleneck dimension 2 as the default setting because it provides a compact autoencoder structure with nearly identical average performance and slightly better cross-attack stability.

11) Computational Overhead: All detection times are measured under the same evaluation environment and averaged over six attacks. The reported time includes trajectory extraction and sample scoring, but does not include the initial training of the target classifier. Table V reports the average detection time of different methods over six attacks. PSSP requires 1.743 minutes on average, which is lower than STRIP, DBR, and TeCo. Specifically, PSSP is about 3.68 times faster than STRIP, 3.50 times faster than DBR, and 43.98 times faster than TeCo. Although SCALE-UP and IBD-PSC have lower detection time, their access assumptions and detection mechanisms are different: SCALE-UP relies on scaled prediction consistency, while IBD-PSC requires parameter or BN-layer access. In contrast, PSSP operates under a soft-label black-box setting and does not access model parameters, gradients, or hidden representations.

The additional memory overhead of PSSP is also limited. The autoencoder used in our experiments is a lightweight fully connected network with the structure $10 \rightarrow 32 \rightarrow 16 \rightarrow 2 \rightarrow$

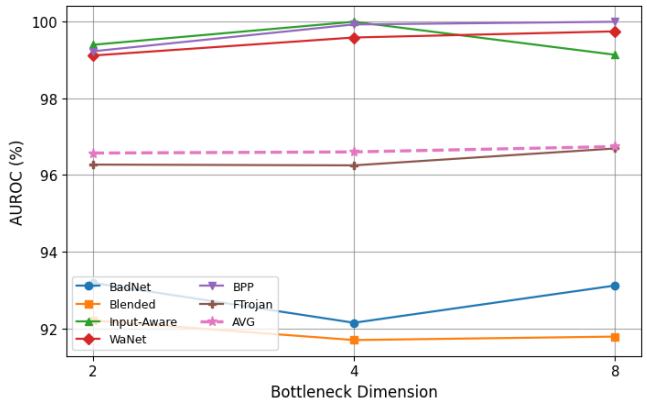

Fig. 9: Sensitivity of PSSP to the autoencoder bottleneck dimension on CIFAR-10 with PreAct-ResNet18. Each curve reports AUROC under one attack type, and AVG denotes the average AUROC across attacks.

TABLE V: Average detection time of different methods over six attacks. The unit is minutes. The reported time includes trajectory extraction and sample scoring, but excludes the initial training of the target classifier.

Method	Detection Time (min)
STRIP	6.410
DBR	6.103
SCALE-UP	**0.375**
IBD-PSC	0.867
TeCo	76.660
PSSP (Ours)	1.743

$16 \to 32 \to 10$, containing only a small number of parameters. During detection, PSSP only stores low-dimensional loss trajectories and reconstruction scores, without saving gradients or hidden-layer features. Therefore, the proposed method remains practical for training-set screening while maintaining strong average detection performance.

12) Defense-Aware Adaptive Attack: To evaluate the robustness boundary of PSSP against defense-aware adversaries, we conduct a trajectory-mimicking adaptive attack experiment following the adaptive evaluation protocol used in [14]. In this experiment, the attacker is assumed to know the proposed detection mechanism, including the progressive perturbation process and the loss-trajectory representation used by the detector. The attack objective is therefore designed to weaken the detector by encouraging poisoned samples to mimic the average loss-trajectory pattern of clean samples.

Specifically, an adaptive loss is introduced during poisoned model training. Given a poisoned mini-batch $\mathscr{B}_p$, the loss trajectory of a poisoned sample x^p under the PSSP perturbation sequence is denoted as $\ell(x^p)$. Let μ_ℓ denote the mean clean loss trajectory estimated from clean samples. The adaptive trajectory-mimicking loss is formulated as

$$\mathscr{L}_{\text{ada}} = \frac{1}{|\mathscr{B}_p|} \sum_{x^p \in \mathscr{B}_p} \|\ell(x^p) - \mu_\ell\|_2^2. \tag{18}$$

TABLE VI: Defense-aware adaptive attack results on CIFAR-10 with PreAct-ResNet18. λ controls the strength of the trajectory-mimicking objective, while BA, ASR, and AUROC denote benign accuracy, attack success rate, and detection performance, respectively.

Attack	λ	BA (%)	ASR (%)	AUROC (%)
BadNet	0	90.29	92.73	93.19
	0.005	70.37	96.67	51.06
	0.01	74.87	89.66	34.65
	0.05	78.78	20.73	6.32
FTrojan	0	92.79	100.00	96.27
	0.005	67.74	27.52	60.84
	0.01	35.84	0.01	12.95
	0.05	75.66	8.85	4.63

By minimizing this term, poisoned samples are encouraged to produce loss trajectories closer to the clean trajectory manifold, thereby simulating a defense-aware attempt to bypass the proposed detector. By incorporating the standard backdoor training objective $\mathscr{L}_{\text{bd}}$, which includes clean classification and targeted poisoned-sample losses, the overall adaptive training objective is formulated as

$$\mathscr{L} = \mathscr{L}_{\text{bd}} + \lambda \mathscr{L}_{\text{ada}}, \tag{19}$$

where λ controls the strength of the adaptive loss. In this way, the backdoor objective is preserved, while the additional adaptive term encourages poisoned samples to mimic the clean loss-trajectory pattern. When $\lambda = 0$, the attack reduces to the non-adaptive setting.

Experiments are conducted on CIFAR-10 with PreAct-ResNet18 under BadNet and FTrojan attacks, representing spatial patch triggers and frequency-domain triggers, respectively. We vary λ and report benign accuracy (BA), attack success rate (ASR), and AUROC in Table VI. BA and ASR measure attack utility, while AUROC measures the detection separability of PSSP.

The results show that the adaptive objective can reduce the detection AUROC, but this reduction is achieved with a clear cost in attack utility. For BadNet, AUROC drops from 93.19% to 51.06% when $\lambda = 0.005$, but BA also decreases sharply from 90.29% to 70.37%. When $\lambda = 0.05$, AUROC further drops to 6.32%, while ASR collapses to 20.73%, indicating that the backdoor attack is no longer effective. A similar trend is observed for FTrojan. AUROC decreases from 96.27% to 60.84% at $\lambda = 0.005$, but ASR drops from 100.00% to 27.52%. When $\lambda = 0.01$, ASR almost vanishes to 0.01%.

These results indicate that defense-aware trajectory mimicry can weaken the detector, but effective evasion requires sacrificing benign accuracy or attack success rate. These results do not imply that PSSP is immune to adaptive attacks. Instead, they show that successful trajectory mimicry introduces a clear trade-off between detection evasion and attack utility, which clarifies the robustness boundary of the proposed method.

V. Conclusion

This paper presented a progressive spatial and spectral perturbation (PSSP) framework for backdoor sample detection. The proposed method provides a trajectory-based perspective for analyzing model sensitivity under structured distortions. By jointly applying Gaussian blur and additive Gaussian noise in a progressive manner, PSSP weakens trigger-related cues and reveals different response patterns between clean and poisoned samples. Clean samples generally show smoother loss changes due to their reliance on semantic features, whereas poisoned samples tend to exhibit larger deviations when trigger-dependent cues are disrupted. By modeling these responses as loss trajectories and using autoencoder reconstruction errors for anomaly detection, PSSP provides an efficient, architecture-independent, and soft-label black-box detection strategy.

The experimental results support several findings. First, progressive perturbation can provide an effective diagnostic signal for identifying poisoned samples without requiring model internals or trigger priors. Second, the spatial-spectral perturbation mechanism combines spatial smoothing and spectrally distributed noise, offering a useful explanation of trigger degradation from both spatial and frequency-domain perspectives. Third, experiments across datasets, architectures, and attack types show that PSSP achieves strong and stable detection performance compared with representative perturbation-based methods and recent input-level defenses, including SCALE-UP and IBD-PSC where applicable. The sensitivity analyses further indicate that PSSP remains stable under reasonable threshold, perturbation, clean-data, and autoencoder settings. The defense-aware adaptive attack experiment also shows that trajectory mimicry can weaken detection performance, but effective evasion introduces a clear trade-off with benign accuracy or attack success rate. These results suggest that transforming perturbation dynamics into loss-trajectory signals is a promising direction for practical backdoor sample detection.

Future work will explore adaptive perturbation scheduling guided by information-theoretic or attention-based mechanisms, enabling the perturbation process to adjust its intensity according to input sensitivity. Extending PSSP to multimodal and sequential data, such as audio, video, and time-series, could further evaluate its applicability beyond image classification. Future work will also examine PSSP under physically realizable backdoor settings, where trigger appearance may be affected by printing quality, camera noise, lighting, and viewpoint changes. Another practical direction is to integrate PSSP into automated data sanitization or model auditing pipelines for continuous monitoring of model integrity during deployment. In addition, reducing the dependence on trusted clean calibration data and developing stronger defense-aware evaluations remain important directions for improving the reliability of trajectory-based backdoor detection.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[3] Z. Zou, K. Chen, Z. Shi *et al.*, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023.

[4] T. Gu, B. Dolan-Gavitt, and S. Garg, "Badnets: Identifying vulnerabilities in the machine learning model supply chain," *arXiv preprint arXiv:1708.06733*, 2017.

[5] Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang, and X. Zhang, "Trojaning attack on neural networks," 2017.

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

[7] Z. Wang, J. Zhai, and S. 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*, 2022, pp. 15 074–15 084.

[8] A. Nguyen and A. Tran, "Wanet–imperceptible warping-based backdoor attack," *arXiv preprint arXiv:2102.10369*, 2021.

[9] T. Wang, Y. Yao, F. Xu, S. An, H. Tong, and T. Wang, "An invisible black-box backdoor attack through frequency domain," in *17th European Conference on Computer Vision, ECCV 2022*. Springer, 2022, pp. 396–413.

[10] M. Zhu, Y. Li, J. Guo, T. Wei, S.-T. Xia, and Z. Qin, "Towards sample-specific backdoor attack with clean labels via attribute trigger," *IEEE Transactions on Dependable and Secure Computing*, vol. 22, no. 5, pp. 4685–4698, 2025.

[11] L. Hou, Z. Hua, Y. Li *et al.*, "M-to-n backdoor paradigm: A multi-trigger and multi-target attack to deep learning models," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 11, pp. 11 299–11 312, 2024.

[12] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "Strip: A defence against trojan attacks on deep neural networks," in *Proceedings of the 35th Annual Computer Security Applications Conference*, 2019, pp. 113–125.

[13] W. Chen, B. Wu, and H. Wang, "Effective backdoor defense by exploiting sensitivity of poisoned samples," *Advances in Neural Information Processing Systems*, vol. 35, pp. 9727–9737, 2022.

[14] X. Liu, M. Li, H. Wang *et al.*, "Detecting backdoors during the inference stage based on corruption robustness

consistency," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 363–16 372.

[15] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao, "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 707–723.

[16] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *International Symposium on Research in Attacks, Intrusions, and Defenses*. Springer, 2018, pp. 273–294.

[17] Y. Li, X. Lyu, N. Koren *et al.*, "Neural attention distillation: Erasing backdoor triggers from deep neural networks," *arXiv preprint arXiv:2101.05930*, 2021.

[18] D. Wu and Y. Wang, "Adversarial neuron pruning purifies backdoored deep models," *Advances in Neural Information Processing Systems*, vol. 34, pp. 16 913–16 925, 2021.

[19] Y. Li, X. Lyu, X. Ma *et al.*, "Reconstructive neuron pruning for backdoor defense," in *International Conference on Machine Learning*. PMLR, 2023, pp. 19 837–19 854.

[20] J. Guo, Y. Li, X. Chen, H. Guo, L. Sun, and C. Liu, "SCALE-UP: An efficient black-box input-level backdoor detection via analyzing scaled prediction consistency," in *International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=o0LFPcoFKnr

[21] L. Hou, R. Feng, Z. Hua, W. Luo, L. Y. Zhang, and Y. Li, "IBD-PSC: Input-level backdoor detection via parameter-oriented scaling consistency," in *Proceedings of the 41st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 235. PMLR, 2024, pp. 18 992–19 022. [Online]. Available: https://proceedings.mlr.press/v235/hou24a.html

[22] G.-Q. Zeng, J.-M. Shao, K.-D. Lu, G.-G. Geng, and J. Weng, "MoCC-BD-FID: Multi-objective clustering combination-based backdoor defense for federated intrusion detection of industrial control systems," *IEEE Transactions on Information Forensics and Security*, vol. 20, pp. 6868–6883, 2025.

[23] G.-Q. Zeng, H.-N. Wei, K.-D. Lu, G.-G. Geng, and J. Weng, "DACO-BD: Data augmentation combinatorial optimization-based backdoor defense in deep neural networks for SAR image classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–13, 2024.

[24] J. Guan, J. Liang, and R. He, "Backdoor defense via test-time detecting and repairing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 24 564–24 573.

[25] Y. Li, Y. Li, B. Wu *et al.*, "Invisible backdoor attack with sample-specific triggers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 16 463–16 472.

[26] T. A. Nguyen and A. Tran, "Input-aware dynamic backdoor attack," *Advances in Neural Information Processing Systems*, vol. 33, pp. 3454–3464, 2020.

[27] K. Doan, Y. Lao, W. Zhao, and P. Li, "Lira: Learnable, imperceptible and robust backdoor attacks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 966–11 976.

[28] A. Salem, R. Wen, M. Backes, S. Ma, and Y. Zhang, "Dynamic backdoor attacks against machine learning models," in *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2022, pp. 703–718.

[29] W. Jiang, H. Li, G. Xu *et al.*, "Color backdoor: A robust poisoning attack in color space," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 8133–8142.

[30] Y. Li, X. Ma, J. He *et al.*, "Multi-trigger backdoor attacks: More triggers, more threats," *CoRR*, 2024.

[31] Z. Li, P. Li, X. Sheng *et al.*, "Imtm: Invisible multi-trigger multimodal backdoor attack," in *CCF International Conference on Natural Language Processing and Chinese Computing*. Cham: Springer Nature Switzerland, 2023, pp. 533–545.

[32] G. Singh, J. Kumar, and R. K. Mohapatra, "Physical backdoor attack using multi-trigger to same class," in *2023 3rd International Conference on Artificial Intelligence and Signal Processing (AISP)*. IEEE, 2023, pp. 1–5.

[33] B. Tran, J. Li, and A. Madry, "Spectral signatures in backdoor attacks," *Advances in neural information processing systems*, vol. 31, 2018.

[34] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[35] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The german traffic sign recognition benchmark: a multi-class classification competition," in *The 2011 international joint conference on neural networks*. IEEE, 2011, pp. 1453–1460.

[36] B. Wu, H. Chen, M. Zhang *et al.*, "Backdoorbench: A comprehensive benchmark of backdoor learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 10 546–10 559, 2022.

[37] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," *Advances in Neural Information Processing Systems*, vol. 28, 2015.