



AFI: Blackbox Backdoor Detection Method Based on Adaptive Feature Injection

Simin Tang^{1,2,3,4}, Zhiyong Zhang^{1,2,3,4,*}, Junyan Pan^{1,2,3,4}, Gaoyuan Quan^{1,2,3,4}, Weiguo Wang⁵ and Junchang Jing⁶

¹Information Engineering College, Henan University of Science and Technology, Luoyang, 471023, China

²Henan International Joint Laboratory of Cyberspace Security Applications, Henan University of Science and Technology, Luoyang, 471023, China

³Henan Intelligent Manufacturing Big Data Development Innovation Laboratory, Henan University of Science and Technology, Luoyang, 471023, China

⁴Institute of Artificial Intelligence Innovations, Henan University of Science and Technology, Luoyang, 471023, China

⁵Education Technology Department, New H3C Technologies Co., Ltd., Beijing, 100102, China

⁶College of Computer and Information Engineering, Henan Normal University, Xinxiang, 453007, China

*Corresponding Author: Zhiyong Zhang. Email: xidianzzy@126.com

Received: 25 September 2025; Accepted: 05 December 2025; Published: 10 February 2026

ABSTRACT: At inference time, deep neural networks are susceptible to backdoor attacks, which can produce attacker-controlled outputs when inputs contain carefully crafted triggers. Existing defense methods often focus on specific attack types or incur high costs, such as data cleaning or model fine-tuning. In contrast, we argue that it is possible to achieve effective and generalizable defense without removing triggers or incurring high model-cleaning costs. From the attacker's perspective and based on characteristics of vulnerable neuron activation anomalies, we propose an Adaptive Feature Injection (AFI) method for black-box backdoor detection. AFI employs a pre-trained image encoder to extract multi-level deep features and constructs a dynamic weight fusion mechanism for precise identification and interception of poisoned samples. Specifically, we select the control samples with the largest feature differences from the clean dataset via feature-space analysis, and generate blended sample pairs with the test sample using dynamic linear interpolation. The detection statistic is computed by measuring the divergence $G(x)$ in model output responses. We systematically evaluate the effectiveness of AFI against representative backdoor attacks, including BadNets, Blend, WaNet, and IAB, on three benchmark datasets: MNIST, CIFAR-10, and ImageNet. Experimental results show that AFI can effectively detect poisoned samples, achieving average detection rates of 95.20%, 94.15%, and 86.49% on these datasets, respectively. Compared with existing methods, AFI demonstrates strong cross-domain generalization ability and robustness to unknown attacks.

KEYWORDS: Deep learning; backdoor attacks; universal detection; feature fusion; backward reasoning

1 Introduction

In recent years, deep learning [1] has been widely applied across fields such as image classification, natural language processing, and pattern recognition, becoming a major focus of research in artificial intelligence. Given a test image, its predicted category can be obtained by computing the similarity between the image features and the textual features of category descriptions. However, as neural networks grow increasingly complex—larger parameter scales and deeper architectures—accuracy may improve, but robustness often decreases, introducing additional security vulnerabilities. As a result, neural networks become inherently susceptible to backdoor attacks [2].



Backdoor attacks [3], which constitute a highly covert security threat, rely on the core mechanism of injecting specific malicious behavior patterns into the model through training data poisoning. In such attacks, attackers first carefully design backdoor triggers, which may be particular pixel patterns in the image (such as local color blocks used by BadNets [2]), special word sequences in natural language (such as specific combinations of harmless words as shown in [4]), or specific data features across modalities (such as frequency domain perturbations proposed in [5]). Subsequently, attackers implicitly establish associations between triggers and target outputs during the learning process by contaminating the training dataset (typically, only 1%–5% of the data needs to be contaminated [2]). A key characteristic of this attack is that the model implanted with a backdoor performs similarly to a clean model under normal input (the difference in test accuracy is usually less than 0.5%). Still, once the input contains preset triggers (such as specific pixel combinations in image corners or special character sequences in text), the model will perform malicious actions predetermined by the attacker, such as misclassifying any input into the target category, generating harmful content, and even leaking private data. More seriously, such triggers can be imperceptible to the human eye and have minimal impact on the normal functioning of the model, posing significant challenges to traditional anomaly-detection-based defense methods. The main process of backdoor attack and defense is shown in Fig. 1.

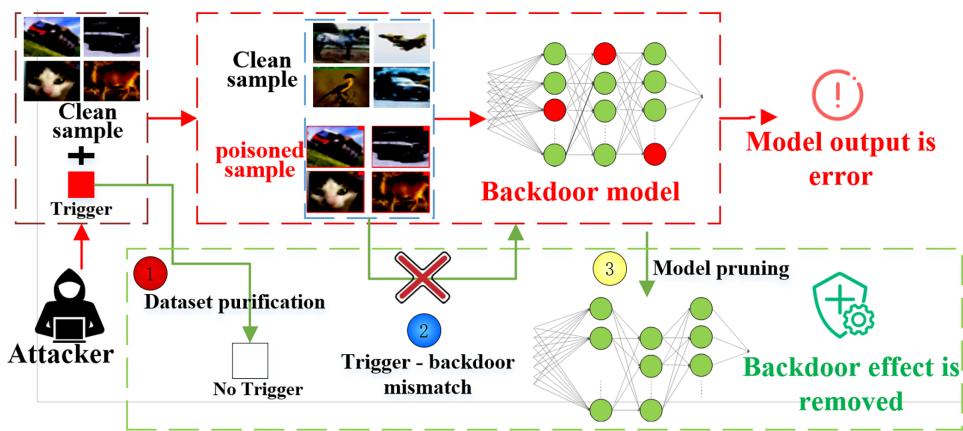


Figure 1: Basic process of backdoor attack and defense

The current research on defense against backdoor attacks in deep neural networks mainly faces two limitations: firstly, most existing defense methods are designed for specific types of backdoor attacks, making it difficult to cope with the constantly evolving and diverse attack methods. Secondly, mainstream defense solutions often require high computational costs, including but not limited to: (1) global parameter adjustment or retraining of pre-trained models [6], which can require thousands of GPU hours for large models such as ViT-Huge; and (2) pruning defective neurons [7], which demands a large number of clean samples that are difficult to obtain in practice. Such targeted and computationally intensive properties severely restrict the practical deployment of defense methods in real-time systems and resource-constrained environments.

To address the above issue, we select multiple pairs of reference samples with maximal mutual differences from the clean dataset based on feature space analysis. We use dynamic linear interpolation to generate mixed sample pairs with the test sample and construct detection statistics by observing the model output response dispersion $G(x)$. The ability to induce misclassification errors is a general characteristic of backdoored samples. Therefore, the proposed defense is attack-agnostic [6], which distinguishes it from existing defenses.

The main contributions are as follows:

- 1) We propose a general black-box backdoor detection method based on a hybrid injection strategy. This method fuses the input sample under inspection with multiple clean samples that have significant differences in their features from each other, and infers whether the sample contains a trigger based on the model output on the fused input. It does not rely on the model structure or parameters, and does not require modification of the original inputs or the model itself, which provides good generalization and black-box adaptability.
- 2) We construct a dynamic-weight trigger fusion mechanism. This mechanism utilizes the characteristics of trigger-induced neuron activation value maximization and manipulation of model predictions in backdoor attacks. By comparing and analyzing the model's output responses, it determines whether the input sample is influenced by a trigger, thereby effectively identifying poisoned samples.
- 3) We propose a new defense evaluation metric (Detection Stability Performance, DSP) to verify the cross-domain generalization ability and robustness against unknown attacks of our method. The experimental results show that the proposed method can effectively detect poisoned samples, with average detection rates of 95.2%, 94.15%, and 86.49% on three benchmark datasets, respectively.

The remainder of this article is organized as follows. [Section 2](#) reviews related work on backdoor attacks and defenses. [Section 3](#) presents the design of the proposed AFI defense method. [Section 4](#) reports experimental results demonstrating the effectiveness of AFI. Finally, [Section 5](#) concludes the paper and outlines potential research directions for backdoor defense.

2 Related Work

2.1 Backdoor Attacks

Backdoor attacks typically aim to implant malicious behavior into deep learning models by injecting a small number of poisoned samples into the training dataset. Specifically, once the model is trained on these poisoned samples, it will misclassify the samples into the target class when the trigger is activated. However, when the trigger is not present, the backdoor model behaves the same as a normal model and does not show anomalous behavior. According to the attack method, existing backdoor attacks can be classified into two categories: (1) **poison-label attacks** [8–12], which connect the trigger and the target class by changing the labels of the toxic samples to the target labels to enhance the attack effect [9,13] or to hide the traces of the attack [14–16]. (2) **Clean-label attacks** [17–19], which keep the original labels of the samples unchanged and only poison samples within the target class by injecting triggers. Although clean label attacks are more stealthy, they may sometimes fail to successfully implant a backdoor [19,20].

2.2 Backdoor Defense

Existing backdoor defense methods can be broadly divided into three categories: (1) **Model reconstruction defenses**. These methods directly modify a suspicious model to suppress or remove backdoor behaviors, usually by first synthesizing triggers and then mitigating their influence [6,7,21,22]. This type of method largely relies on the quality of synthesized triggers, so the defense effect may not be satisfactory when facing more complex triggers [23,24]. For instance, Neural Cleaner (NC) [6] generates a trigger for each category and employs Median Absolute Deviation (MAD) for outlier detection, followed by a forgetting strategy to eliminate the backdoor. Zhu et al. [7] use the strong reconstruction capability of GANs to detect and “clean” neural backdoors without requiring access to the training data, showing robustness and efficiency across various settings. A new Trojan network detection mechanism [21] locates a “winning Trojan lottery ticket” that retains almost complete Trojan information but only chance-level performance on clean input,

and then restores the triggers embedded in this already isolated subnetwork. The Shapley Pruning [22] can identify and remove less than 1% of infected neurons while maintaining model structure and accuracy, even with extremely limited data (one or zero samples per class). (2) **Pruning-based defenses.** These approaches aim to detect and prune malicious neurons and typically require access to clean labeled data, which is often impractical in real-world scenarios. For example, Fine-Pruning [25] combines neuron pruning with fine-tuning to suppress or eliminate backdoor behavior. (3) **Input detection defenses.** This category focuses on identifying poisoned samples at inference time without modifying the model. Activation-based defenses such as STRIP [26] detect backdoors by measuring prediction entropy under input perturbations, while SentiNet [27] localizes suspicious regions using model interpretability. MNTD [28] trains a meta-classifier to distinguish clean and Trojaned models from their behavior. Defenses can be categorized by the defender's knowledge into sample-, model-, and training data-level approaches [29]. For example, CCA-UD [29] operates at the training data level by clustering samples to identify clean and poisoned ones, which is computationally expensive. In contrast, our method achieves high-level defense directly at the sample level.

3 Our Method: Fusion Triggered Detection (AFI)

3.1 Basic Settings

Attack Setting: In backdoor attacks on classification tasks, we train a DNN model $C_\theta: \mathcal{X} \rightarrow \mathcal{Y}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ denotes the input space and $\mathcal{Y} = \{1, 2, \dots, K\}$ denotes the set of class labels. The training dataset is denoted by $D_{\text{train}} = \{(x_n, y_n)\}_{n=1}^N$, where $x_n \in \mathcal{X}$ is the n -th input sample and $y_n \in \mathcal{Y}$ is its corresponding class label. For a clean input x_n , the model output satisfies $C_\theta(x_n) = y_n$, indicating that the prediction is correct on normal samples. For an input sample $x' = x + t(x)$ with a backdoor trigger $t(x)$, the model satisfies $C_\theta(x') = y_t$, where y_t is the target label specified by the attacker. To implement the backdoor attack, αN (where $0 < \alpha < 1$) samples are selected from the training dataset as poisoned samples, and the following modifications are applied: (1) **Label modification:** change the labels of the poisoned samples to the target label y_t . (2) **Trigger injection:** add a trigger $t(x)$ to each poisoned sample so that $x' = x + t(x)$ becomes a triggered sample. After processing the samples, we train the model C_θ on the modified training set so that it behaves normally on clean samples while being activated by the embedded triggers for backdoor attacks.

Defense Setting: This study considers a realistic black-box defense scenario, where the defender obtains a pre-trained model from an untrusted source without access to its training process, and is only given clean samples and test samples for inspection. (1) **Defense Objective:** Detect poisoned samples with backdoor triggers while maintaining accuracy on clean samples comparable to standard defenses. (2) **Defense Capability:** Our method detects backdoored samples without modifying the dataset or model, preventing trigger-target associations. It generalizes to multiple types of backdoors and various image classification datasets without requiring knowledge of model architecture or parameters.

3.2 Defense Model

Our backdoor defense adopts the inverse reasoning concept of “input-output” to achieve accurate sample detection. Firstly, a convolutional neural network (CNN) extracts feature vectors for all images in the dataset. By calculating the difference between feature vectors, we select the two clean images with the largest difference in features between them. Secondly, the test image is fused with each of the two most dissimilar images at a certain ratio. Finally, the two mixed samples are input into the model to obtain two outputs, and the consistency of these outputs is used to determine whether the input samples are poisoned, thereby achieving backdoor detection. The specific detection flowchart is shown in Fig. 2.

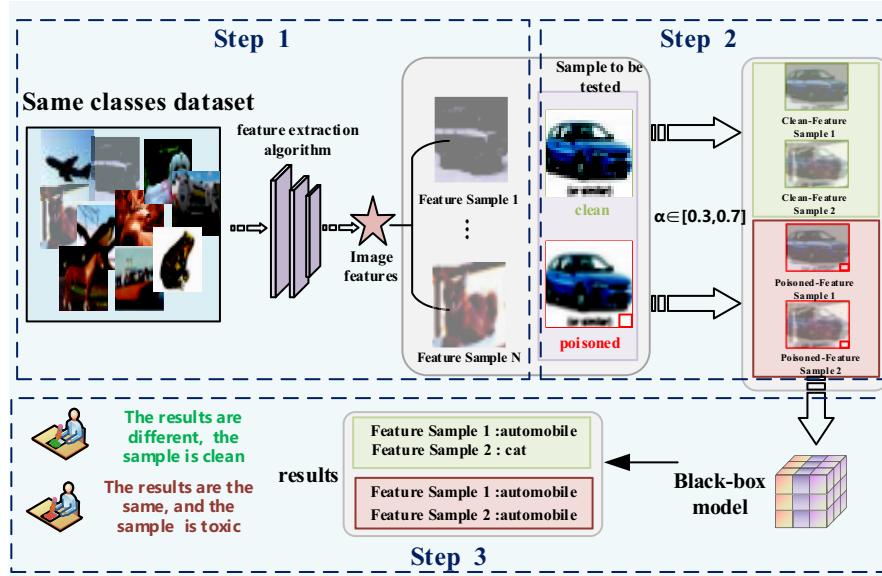


Figure 2: AFI process diagram

It is worth noting that in order to carry out effective backdoor attacks, attackers must carefully design the spatial location and pattern of triggers. Attackers typically maximize the mask of the trigger with the activation value of one or more neurons to create trigger-sensitive internal neurons. This design reinforces the activation of vulnerable neurons and establishes a strong association between the trigger and these neurons. As a result, once the model detects the trigger, these neurons are strongly activated, leading to a corresponding misclassification.

From a defense perspective, this characteristic can be exploited to identify poisoned inputs. The conceptual basis of AFI is as follows: poisoned samples contain a backdoor trigger that is specifically designed to maximize the activation of a small set of neurons highly sensitive to the trigger, establishing a strong association between the trigger and these neurons. Once the model detects the trigger, these neurons are significantly activated and dominate the output. Even when a poisoned sample is blended with other images, this dominant activation persists, resulting in consistently predicted labels. In contrast, clean samples lack such dominant features, so blending introduces variability in their representations, leading to unstable predictions. This difference in prediction stability under blending provides the conceptual foundation for AFI's ability to distinguish poisoned samples from clean ones. When the input contains a trigger, it causes significant increases in neuron activations, which heavily influence the model output. Therefore, when the poisoned sample is fused with the clean sample, the model output will still be affected by the triggers and each output matches the attacker-specified label specified by the attacker. When the sample to be detected is a clean sample, the model output will be affected by the fusion of the two samples with which it is fused due to the large differences in their characteristics, showing significant differences. As illustrated in Fig. 2.

3.3 AFI

3.3.1 Feature Extraction

In order to successfully extract two image samples with significant feature differences from a clean dataset, we employ the widely used ResNet-18 as a generic feature encoder, denoted as ϕ . This choice is based on the following considerations. First, the ResNet architecture, which effectively mitigates the degradation problem in deep networks through residual connections, is a representative model widely adopted in the

field of computer vision. Its strong performance across diverse tasks, from MNIST to the ImageNet subset, ensures the applicability of our method. Second, to maintain consistency in the experimental framework, the target classifiers C under attack in this study and the feature encoder ϕ used here are both based on the ResNet architecture. This approach aims to control variables and provide a purer evaluation of the defense method's own effectiveness.

Furthermore, we note that the choice of model architecture may affect the specific geometry of the feature space. However, the core mechanism of our method relies on a more universal principle: that backdoor attacks cause poisoned samples to become statistical outliers in the deep feature space. Extensive research has shown that high-level features obtained from pre-training on different CNN architectures (e.g., VGG, ResNet), despite differences in spatial geometry, maintain consistency in their discriminability for semantic content. Therefore, the effectiveness of the extreme disparity sample pairs identified based on ResNet-18 features is expected to transfer to other modern architectures, ensuring the robustness of our method's core conclusions and confirming that it is not limited to a specific network.

The general process and formulas for image feature extraction are outlined as follows.

Firstly, convolutional layers serve as the foundation for image feature extraction. The convolution operation involves sliding a filter (or kernel) over the input image and performing dot products to generate feature maps.

Assuming the input image is denoted as I and the filter weights as W , the convolution operation can be expressed as:

$$O(i, j) = \sum_m \sum_n I(i + m, j + n) W(m, n) + b \quad (1)$$

$O(i, j)$ denotes the value at position (i, j) in the output feature map, $I(i + m, j + n)$ represents the pixel value at the corresponding position in the input image, $W(m, n)$ refers to the weights of the convolution kernel, and b is the bias term.

In order to introduce nonlinearity in the model, an activation function is typically applied after the convolution operation. The most commonly used activation function is the Rectified Linear Unit (ReLU), which is defined as follows:

$$f(x) = \max(0, x) \quad (2)$$

The activation function performs a nonlinear transformation on the output of the convolutional layer.

The pooling layer is used to reduce the size of the feature map while retaining important features. A common pooling operation is maximum pooling with the formula:

$$O(i, j) = \max_{m, n} (I(i + m, j + n)) \quad (3)$$

where $I(i + m, j + n)$ is the pixel value within the pooling window and $O(i, j)$ is the output value after pooling.

At the end of the encoder, the extracted features are usually flattened into a vector, and a fully connected layer is used to generate the final feature encoding. The formula for the fully connected layer is:

$$Y = W \times x + b \quad (4)$$

where x is the input vector (flattened feature map), W is the weight matrix, b is the bias term, and Y is the output feature vector.

The second step is to encode and output features. After multiple layers of convolution, pooling, and fully connected operations, the image is finally encoded into a feature vector h .

Finally, by using Euclidean distance to calculate the difference between the feature vectors, the two images with the largest difference are found, and the formula is as follows:

$$D(h_i, h_j) = \sqrt{\sum_{k=1}^n (h_{i,k} - h_{j,k})^2} \quad (5)$$

where h_i and h_j are the feature vectors of the two images, respectively, $h_{i,k}$ and $h_{j,k}$ are the k -th components in the feature vector, and n is the dimension of the feature vector.

We use Euclidean distance for its simplicity and interpretability. Since our method mainly relies on poisoned samples being statistical outliers in the feature space, the choice of distance metric is not critical, and other reasonable metrics are expected to yield similar results.

After the feature extraction operation using ϕ , the two clean samples with the most disparate feature vectors h_1 and h_2 are identified and denoted as c_1 and c_2 , respectively. The flowchart for selecting feature sample pairs is shown in Fig. 3.

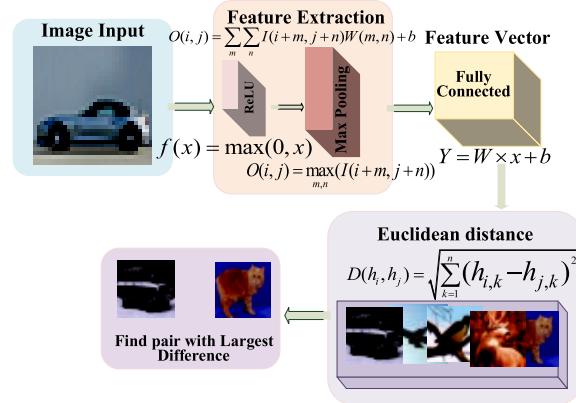


Figure 3: Process diagram for selecting feature sample pairs

3.3.2 Sample Fusion

We adopt the Blended Injection Strategy (BIS) [30]. The image samples to be detected are fused with two clean samples that exhibit significant feature differences, in order to construct hybrid samples with discriminative properties. Specifically, if the original detection image is c and the two feature comparison samples are c_1 and c_2 , the fusion process can be formally expressed as:

$$c_{\text{mix}} = c_i \times (1 - \alpha) + c \times \alpha, \quad i \in \{1, 2\} \quad (6)$$

where $\alpha \in (0, 1)$ represents the image fusion ratio, which controls the blending weights between the original detection image and the two clean feature contrast samples.

3.3.3 Reverse Reasoning

Through the above feature fusion step, two mixed samples can be constructed respectively and recorded as $c_{\text{mix}1}$ and $c_{\text{mix}2}$. Then, these mixed samples are input into the target classifier C to obtain their corresponding prediction outputs $y_1 = C(c_{\text{mix}1})$ and $y_2 = C(c_{\text{mix}2})$. By comparing the classification results

of y_1 and y_2 , we can infer whether the sample is controlled by the potential backdoor trigger. If $y_1 = y_2$, it indicates that the model maintains a stable prediction of a particular category under the fusion disturbance, which means that the sample to be detected contains triggers and thus activates backdoor behavior in the model. On the contrary, if the two prediction results are significantly different, it means that the sample to be detected is normal, and its decision boundary is vulnerable to fusion disturbance. The decision function $G(x)$ is defined as follows:

$$G(x) = \begin{cases} 1, & y_1 = y_2 \\ 0, & y_1 \neq y_2 \end{cases} \quad (7)$$

If $G(x) = 1$, the sample to be detected is suspected to be a poisoned sample. Conversely, it is considered to be a clean sample. This discriminative mechanism does not need to access the model structure and parameters, which is suitable for black-box detection, and the fusion disturbance enlarges the output stability of poisoned samples, thus improving the accuracy and robustness of detection.

3.4 Selection Principle and Strategy for the Fusion Ratio α

The fusion ratio α is a key parameter that balances the influence between the test sample x and the clean reference sample c_i . Extensive experiments show that an effective range for α consistently lies between 0.3 and 0.6. The underlying principle is that α should be large enough to preserve potential trigger signals in the test sample, yet small enough to allow the clean reference to provide a strong and correct semantic context for consistency checking.

Based on an in-depth analysis of the spatial characteristics of triggers from different backdoor attacks, we summarize the following empirical selection strategy:

- **For Local Trigger Attacks:** Such as BadNets and IAB, where the trigger is confined to a small region of the image. We recommend using a lower α value (approximately 0.3–0.5). A smaller fusion ratio helps retain the local trigger pattern in the blended image without overwhelming it with clean-sample semantics.
- **For Global Trigger Attacks:** Such as Blend and WaNet, where the trigger is distributed across the entire image as subtle perturbations or warping. We recommend using a higher α value (approximately 0.4–0.6) to ensure that the global trigger maintains sufficient influence in the fused image.

The above strategy provides a robust guideline for scenarios where the attack type is known or can be inferred. For completely unknown cases, we suggest a simple grid search procedure over $\alpha \in [0.3, 0.6]$ with a step size of 0.01, using a small validation subset to select the optimal value. This protocol has been verified effective in our experiments, ensuring the AFI method's robustness and reproducibility across diverse datasets and attack types.

3.5 Summary

Firstly, we propose a sample-level universal backdoor detection method, which thoroughly embodies the dialectical unity of attack and defense. This approach employs a straightforward mechanism to achieve accurate and efficient defense. Secondly, for the backdoor defender, the method only requires a clean sample dataset. By extracting features and computing the feature distances, it selects the two samples with the most significant feature discrepancy. Compared with conventional backdoor defense techniques, our method greatly reduces the amount of data computation and storage. For users, this method does not need any operation on the dataset, just upload and input the image samples to be detected, which is convenient to use. Thirdly, our method focuses on the new defense concept of trigger detection. It only needs to detect the

poisoned samples containing triggers and prevent the toxic samples from entering the model, which not only avoids backdoor attacks but also retains the availability of the model on clean samples. The AFI algorithm is shown in Algorithm 1.

Algorithm 1: Adversarial feature fusion for backdoor detection (AFI)

Input: Test image x ; clean sample set \mathcal{C} ; feature extractor $\phi(\cdot)$; blending ratio $\alpha \in (0, 1)$;

distance function $\text{dist}(\cdot, \cdot)$

Output: Detection result $G(x) \in \{0, 1\}$

Step 1: Reference Sample Selection

Select the most dissimilar clean sample pair from \mathcal{C} :

$$(c_1, c_2) \leftarrow \arg \max_{c_i, c_j \in \mathcal{C}} \text{dist}(\phi(c_i), \phi(c_j))$$

Step 2: Feature Fusion

Generate blended images by combining x with each reference:

$$x_{\text{blend},1} \leftarrow \alpha \cdot x + (1 - \alpha) \cdot c_1$$

$$x_{\text{blend},2} \leftarrow \alpha \cdot x + (1 - \alpha) \cdot c_2$$

Step 3: Consistency Evaluation

Obtain predictions for both blended images using the target classifier $C(\cdot)$:

$$y_1 \leftarrow C(x_{\text{blend},1})$$

$$y_2 \leftarrow C(x_{\text{blend},2})$$

Step 4: Detection Decision

if $y_1 = y_2$ **then**

return 1; *// Flag as poisoned sample*

else

return 0; *// Flag as clean sample*

4 Experiments

4.1 Datasets and Models

The experiment involves three datasets, including MNIST [31], CIFAR-10 [32] and ImageNet subsets [33]. These datasets cover a variety of recognition tasks, including the classification of common objects, fine-grained classification, and action recognition.

All datasets are initially composed of clean samples. We implement four prominent backdoor attacks on these datasets: BadNets, Blend [14], WaNet [16], and IAB [13]. The ResNet-18 architecture [34] serves as the target model for both backdoor attack implantation and defense evaluation.

Attack baseline: we have carried out four common backdoor attacks, including BadNets [9], Blend, WaNet, IAB. In BadNets, WaNet, and IAB, we choose 1 ($y_t = 1$) as the target label, and in Blend, we choose 0 ($y_t = 0$) as the target label. The poisoning rate for all attacks is set to 10%, meaning 10% of the training samples are poisoned. For poison-label attacks (BadNets, WaNet, IAB), this involves modifying both the sample and its label. For the clean-label attack (Blend), only the sample is modified while its original label is preserved.

Defense baseline: We compare our method with six existing backdoor defense methods, including fine pruning (FP) [25], neural attention distillation (NAD) [35], anti-backdoor learning (ABL) [36], STRIP [26], SentiNet [27], and MNTD [28]. Since FP, NAD, and ABL are sensitive to their hyperparameters, we optimize their best results by grid search.

4.2 Evaluation Indicators

In order to evaluate the performance of AFI, we use three indicators: the accuracy of clean images (ACC), the success rate of detecting backdoor images (DSR), and the detection stability (DSP).

ACC reflects the classification ability of the model on a clean image. A higher value indicates that the model is not disturbed by backdoor attacks.

$$ACC_{clean} = \frac{1}{N_{clean}} \sum_{i=1}^{N_{clean}} \prod (C(x_i) = y_i) \quad (8)$$

where N_{clean} is the total number of clean images, $f(x_i)$ is the prediction label of the model for sample x_i , y_i is the real label of sample x_i , and $\prod (\cdot)$ is the indicating function. When the prediction is correct, it returns 1; otherwise, it returns 0.

DSR reflects the ability of AFI to successfully detect whether the sample is poisoned. Ideally, we hope that the defense success rate is as close as possible to 1, which means that the defense can effectively distinguish between clean samples and poisoned samples.

$$DSR = \frac{1}{2} \left[\frac{1}{N_{clean}} \sum_{i=1}^{N_{clean}} \prod (C_i(C_{mix1}) \neq C_i(C_{mix2})) + \frac{1}{N_{poisoned}} \sum_{j=1}^{N_{poisoned}} \prod (C_j(C_{mix1}) = C_j(C_{mix2})) \right] \quad (9)$$

where N_{clean} is the total number of clean images without embedded backdoor triggers, $N_{poisoned}$ is the total number of images with backdoor triggers, The predictions $C(C_{mix1}^{(i)})$ and $C(C_{mix2}^{(i)})$ are obtained by fusing the i -th clean image with the two reference samples. $C(C_{mix1}^{(j)})$ and $C(C_{mix2}^{(j)})$ represent the prediction labels from blending the j -th poisoned image with the two reference samples. $\prod (\cdot)$ is the indicator function. In the clean sample detection calculation, if the prediction labels of two fused images are different, it returns 1; otherwise, it returns 0. In the calculation of poisoned sample detection, if the prediction labels of two fused images are the same, it returns 1; otherwise, it returns 0. The detection success rate DSR of the AFI is the average of DSR_{clean} and $DSR_{poisoned}$.

We propose DSP (Detection Stability Performance) as a new evaluation metric to assess the universality and robustness of detection methods under practical conditions. DSP measures performance consistency across different datasets and attack methodologies. This metric addresses the real-world challenge where users cannot predetermine the attack presence or specific attack types, thus requiring defenses that remain effective in cross-domain scenarios.

DSP combines performance metrics across datasets and attacks, including average clean accuracy (DSP_{ACC}) and average detection rate on poisoned samples (DSP_{DSR}), formulated as:

$$DSP = \frac{1}{2} \left[\sqrt{\frac{1}{n \times m} \sum_{i=1}^n \sum_{j=1}^m ACC_{i,j}} + \sqrt{\frac{1}{n \times m} \sum_{i=1}^n \sum_{j=1}^m DSR_{i,j}} \right] \quad (10)$$

The square root operation normalizes magnitude differences across heterogeneous datasets and attack types, preventing any single condition from dominating the overall DSP. The equal weighting balances the contributions of classification accuracy and detection success rate, which may differ in scale or importance depending on the scenario. These design choices enhance the stability and interpretability of DSP while maintaining sufficient sensitivity to distinguish between methods.

The formula highlights how the square root reduces the influence of extreme values, and the weighting balances the two components, yielding a robust and interpretable evaluation metric across diverse datasets and attack types.

A larger DSP indicates that the detection method is more consistent and stable across scenarios, making it a valuable measure of universality and deployment potential in complex, open environments.

4.3 Result

To verify the superiority of AFI, we conduct four types of backdoor attacks across three datasets using ResNet. While maintaining a high accuracy and attack rate, the results of using AFI in turn are shown in [Table 1](#).

Table 1: Defense-effect of AFI detection

Dataset		Attack method				DSP	
		BadNets	Blend	WaNet	IAB		
MNIST	Before	ACC	98.2	99.1	97.8	96.5	/
		DSR	0.9	0	0.5	0.1	/
	After	ACC	97.1	99.1	96.5	94.7	95.20
		DSR	93.5	94.6	92.0	94.1	
CIFAR-10	Before	ACC	94.9	94.1	93.6	94.2	/
		DSR	0	1.7	0.1	0	/
	After	ACC	94.1	94.0	93.4	93.9	94.15
		DSR	94.5	93.8	94.5	95.1	
ImageNet	Before	ACC	79.5	82.5	79.1	78.2	/
		DSR	0.2	0.5	1.1	0.4	/
	After	ACC	78.1	81.9	78.5	76.5	86.49
		DSR	96.3	92.4	93.3	94.9	

Note: ACC: Accuracy (%); DSR: Detection Success Rate (%).

Evaluation results in [Table 1](#) confirm AFI's universal defense capability. The method elevates DSR from near-zero to >92% across all tested configurations while preserving ACC within 2% of its original value. Consistent DSP performance (86.49–95.20) demonstrates robustness against diverse attack patterns (BadNets, Blend, WaNet, IAB) and dataset complexities, establishing AFI as an effective sample-level backdoor detection solution.

4.4 Comparative

4.4.1 Comparison of Similar Methods

In order to verify the effectiveness of AFI, we select six representative and widely used backdoor defenses as comparison baselines, namely FP, NAD, ABL, STRIP, SentiNet, and MNTD. We systematically evaluate four typical backdoor attacks on the CIFAR-10 dataset, which are BadNets, Blend, WaNet, and IAB. [Table 2](#) summarizes the model classification ACC and DSR under different defense settings on the CIFAR-10 dataset.

Table 2: Comparison of different defense effects on CIFAR-10

Dataset	Attack	CIFAR-10				DSP
		BadNets	Blend	WaNet	IAB	
No defense	ACC	94.9	94.1	93.6	94.2	94.20
	DSR	0	1.7	0.1	0	0.45
FP	ACC	93.9	92.9	90.4	89.3	91.63
	DSR	98.2	<u>22.9</u>	<u>1.4</u>	<u>1.9</u>	31.10
NAD	ACC	88.2	85.8	<u>71.3</u>	82.8	82.03
	DSR	95.4	96.6	93.3	95.8	95.28
ABL	ACC	93.8	91.9	<u>84.1</u>	93.4	90.80
	DSR	98.9	98.4	97.8	94.9	94.15
STRIP	ACC	94.6	89.3	<u>88.7</u>	91.2	90.95
	DSR	98.5	86.2	<u>80.4</u>	92.8	89.48
SentiNet	ACC	93.1	85.7	86.4	91.8	89.25
	DSR	97.6	<u>79.5</u>	<u>81.4</u>	86.3	86.20
MNTD	ACC	93.9	92.8	<u>92.3</u>	89.1	92.03
	DSR	97.2	96.5	93.6	<u>86.3</u>	93.40
AFI (ours)	ACC	94.1	94.0	93.4	93.9	93.85
	DSR	94.5	93.8	94.5	95.1	94.15

Note: Bold data means excellent performance. Underlined data indicates poor performance.

The experimental results show that all attacks achieve high classification accuracy (>93%) in the absence of defense, while the DSR remains nearly zero. This indicates that the model is successfully implanted with a backdoor yet still maintains strong clean-sample performance. FP achieves a high DSR against BadNets, but its defensive capability drops sharply when facing more complex and covert attacks such as Blend, WaNet, and IAB. Although NAD attains a 97.8% defense success rate on WaNet, this comes at the cost of nearly a 20% reduction in accuracy, significantly compromising the original model performance. ABL provides stronger defense across the four attacks, but still suffers from a notable accuracy drop under WaNet, suggesting limitations in handling warped triggers. STRIP and SentiNet, as representative inference-time detection approaches, exhibit considerable performance variability across different attack paradigms. STRIP achieves excellent performance against BadNets (DSR: 98.5%), yet its detection rate decreases substantially on WaNet (DSR: 80.4%), highlighting the inherent limitations of its input-perturbation-based entropy analysis when confronted with global or feature-space perturbations. SentiNet also displays strong attack-dependent behavior, achieving only 79.5% detection on Blend due to the failure of its saliency-based spatial localization assumptions under globally distributed triggers. MNTD, which adopts a meta-learning framework, shows relatively more stable performance with an average DSR of 93.40%. However, its detection rate on IAB (86.3%) is comparatively lower, and its average ACC (92.03%) is also somewhat affected. These results indicate that the method's effectiveness still depends on the diversity and coverage of attack patterns used during training.

In contrast, the proposed AFI method demonstrates consistently strong performance across all attack scenarios. It achieves a high detection rate (94.45% DSR) while preserving excellent classification accuracy

(93.85% ACC), nearly matching the undefended baseline. Compared with inference-time detectors, AFI improves DSR by more than 13 percentage points on complex attacks and avoids the need for meta-training. These results confirm that AFI achieves an optimal balance between detection effectiveness, model preservation, and generalization capability.

[Table 3](#) reports the ACC and DSR of models trained on ImageNet under different defense methods (FP, NAD, DBD, AFI) against four typical backdoor attacks (BadNets, Blend, WaNet, IAB).

Table 3: Comparison of different defense effects on ImageNet

Dataset	Attack	ImageNet				DSP
		BadNets	Blend	WaNet	IAB	
No defense	ACC	79.5	82.5	79.1	78.2	79.83
	DSR	0.2	0.5	1.1	0.4	0.55
FP	ACC	70.3	63.4	58.2	58.7	62.65
	DSR	98.4	90.5	<u>15.6</u>	<u>15.8</u>	55.08
NAD	ACC	<u>65.1</u>	<u>64.8</u>	<u>63.8</u>	<u>63.8</u>	64.38
	DSR	94.9	99.7	98.7	99.4	98.18
ABL	ACC	81.9	82.3	80.6	83.1	81.98
	DSR	99.7	<u>0</u>	90.2	100	72.48
AFI (ours)	ACC	78.1	81.9	78.5	76.5	78.80
	DSR	96.3	92.4	93.3	94.9	86.49

Note: Bold data means excellent performance. Underlined data indicates poor performance.

The experimental results show that, under baseline conditions without defense, all four backdoor attacks maintain high classification accuracy (above 78%), while the DSR remains close to 0. This indicates that the attacks successfully manipulate model behavior with minimal impact on clean-sample classification. FP demonstrates good defensive performance against BadNets and Blend, but performs poorly against more stealthy attacks. For example, the ACC under WaNet and IAB drops to around 15%, indicating that FP is limited and cannot generalize across diverse attack types. NAD achieves consistently high DSR across all attacks, but at the cost of significantly reducing the model's classification accuracy. DBD exhibits highly polarized performance, achieving 100% defense success under IAB but failing (0% DSR) against Blend. In contrast, AFI maintains stable and balanced performance, preserving both high ACC and robust DSR across all attack scenarios.

AFI demonstrates strong robustness, good transferability, and controllable accuracy degradation when defending against diverse backdoor attacks, highlighting its potential for practical deployment. [Fig. 4](#) compares the performance of AFI with three baseline defenses under various backdoor attacks on the ImageNet and CIFAR-10 datasets. As shown in [Fig. 4a](#), both FP and AFI maintain high and stable classification accuracy across the four attacks. However, [Fig. 4b](#) indicates that FP's DSR fluctuates substantially, revealing its instability across different attack types. Similarly, [Fig. 4c](#) shows that DBD and AFI perform well overall, but [Fig. 4d](#) reveals that DBD completely fails against the Blend attack, with its detection success rate dropping to 0%. Overall, AFI achieves more robust and reliable defensive performance across attacks and datasets, outperforming the comparative baselines.

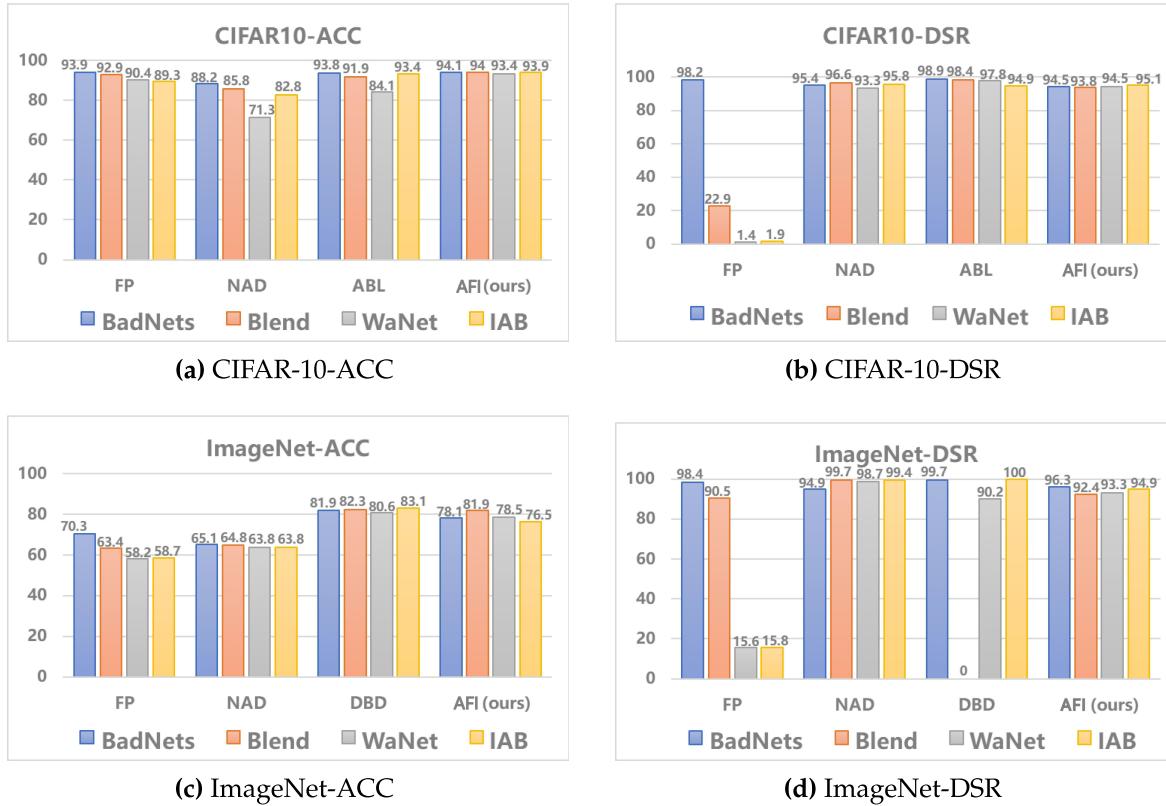


Figure 4: Comparison of indicators based on different datasets

4.4.2 Comparison of Different Fusion Proportions

It should be noted that the optimal fusion ratio of AFI varies across different attack types, generally falling between 0.3 and 0.6, which is a key factor influencing detection performance. (1) For BadNets, the trigger is local and typically resides in background regions. Thus, even when the fused image occupies a large proportion, the trigger remains identifiable. Increasing the fusion ratio improves clean-sample detection while still preserving the trigger in poisoned images. (2) In Blend, the trigger is globally embedded across the entire image, and the target-object region of the fused image introduces interference. Therefore, the fusion ratio must be reduced to maintain a high proportion of the poisoned image and preserve attack effectiveness. (3) Similarly, WaNet embeds its trigger globally via geometric warping, requiring the original image to retain a dominant proportion during fusion to prevent the trigger from being suppressed. (4) For IAB, the trigger is placed in non-object regions, and a larger fusion ratio is suitable, which is consistent with the behavior observed in BadNets.

This section examines the factors affecting the fusion ratio and provides representative results in Fig. 5. Fig. 5a–c respectively shows the defensive accuracy of AFI against BadNets and Blend on MNIST and CIFAR-10.

Comparing Fig. 5a and b, it can be observed that when BadNets is applied to MNIST, the DSR of AFI on clean samples decreases as the fusion ratio increases, while the DSR on poisoned samples increases. The optimal fusion ratio in this setting is 0.3. When BadNets is performed on CIFAR-10, the optimal fusion ratio becomes 0.52. These results indicate that for the same attack across different datasets, the optimal fusion ratio of AFI varies. Fig. 5c shows the results of Blend on MNIST, where the optimal fusion ratio is 0.58. Combined with Fig. 5a and c, it can be concluded that AFI also requires different optimal fusion ratios for different

attack types on the same dataset. Therefore, when facing unknown attack types or datasets, a grid search over the range of 0.3–0.6 (with a step size of 0.01) can be used to identify the optimal fusion ratio.

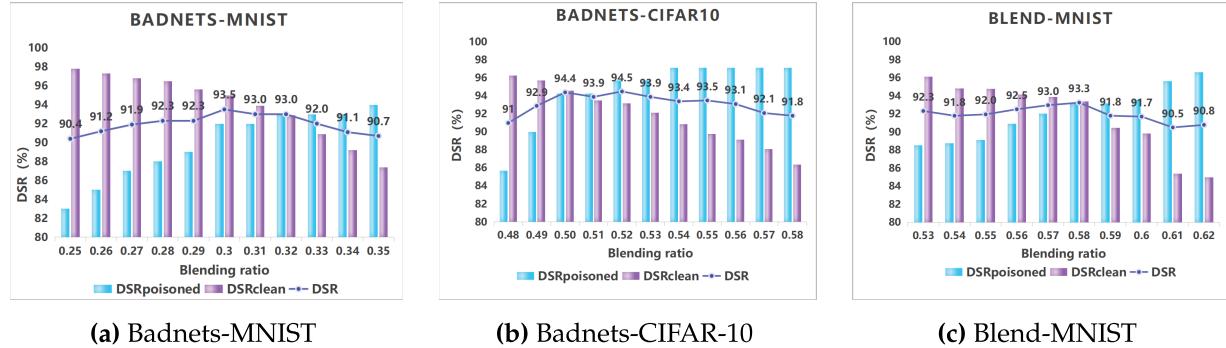


Figure 5: Impact of different fusion ratios on model performance

Taking the experiment in [Fig. 5](#) as an example, the optimal fusion ratio corresponding to [Fig. 5a](#) is 0.3. The optimal fusion ratio corresponding to [Fig. 5b](#) is 0.52, and the visualized image samples with different fusion ratios are shown in [Fig. 6a](#) and [b](#), respectively. Poisoned samples are generated by applying BadNets to clean samples. Clean samples and poisoned samples serve as mutual control groups. Regardless of whether the sample to be tested is toxic or not, when the fusion ratio $\alpha < 0.3$, the two results are identical; When the fusion ratio $\alpha > 0.3$, the two results differ. It is impossible to determine whether the sample under detection is poisoned based on the results. In [Fig. 6a](#), when the fusion ratio $\alpha = 0.3$, if the sample to be detected is clean, $result1 \neq result2$; If the sample to be tested is poisoned, $result1 = result2$. Therefore, we first use a large amount of data in the dataset to test the optimal fusion ratio, and then infer whether the sample to be tested is toxic by judging whether $result1$ is equal to $result2$. [Fig. 6b](#) shows that when $\alpha < 0.52$, the classification results of the two fused images are different, and when $\alpha > 0.52$, the classification results of the two fused images are the same, making it impossible to determine the poisoning status of the test sample based on the classification results.

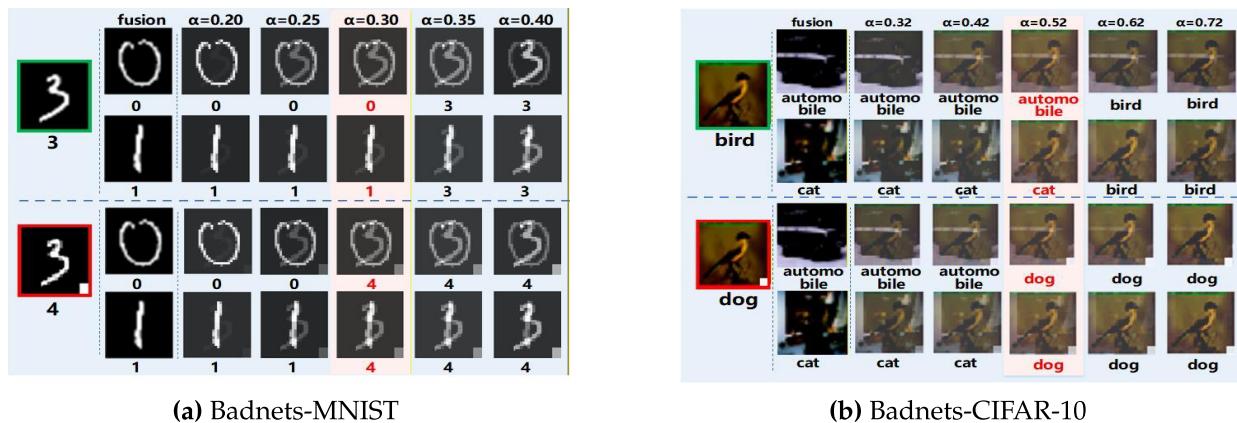


Figure 6: Comparison of model outputs under different fusion ratios

4.5 Robustness Analysis against Adaptive Attackers

This section discusses the theoretical robustness of the AFI method when facing a strong, adaptive attacker with full knowledge of the defense mechanism, addressing the reviewer's valuable comment.

We consider a white-box attack scenario: the attacker is fully aware of AFI's detection pipeline (including the feature extractor, blending ratio α , and decision rule) and aims to design a trigger t that can evade detection.

The attacker's objective is formalized as a dual-goal optimization problem: the trigger must ensure that not only is the poisoned sample $x' = x + t$ classified as the target label y_t , but also that the blended output $C(c_i \times (1 - \alpha) + x' \times \alpha)$ remains y_t when fused with any clean reference sample c_i from the holdout set \mathcal{C} , thereby breaking the prediction consistency check upon which AFI relies.

Theoretically, we argue that successfully crafting such a trigger is inherently difficult for the adversary. AFI's fusion mechanism forces the attacker to solve a conflicting optimization objective: the trigger must be effective on the original poisoned sample x' while simultaneously maintaining its dominance over the model's prediction even when x' is significantly "diluted" by the semantic content of a clean sample. This effectively necessitates the trigger to possess a global and overwhelmingly strong semantic influence in the feature space.

However, achieving this likely comes at a high cost for the attacker, leading to a critical trade-off:

- **Loss of Stealth:** An overly potent and global trigger required to withstand arbitrary fusion becomes more susceptible to detection via visual inspection or statistical anomaly detection, compromising the fundamental requirement of a stealthy backdoor.
- **Degradation of Model Utility:** Embedding such a powerful backdoor functionality often interferes with the model's normal decision-making process, potentially leading to a noticeable drop in clean data accuracy (ACC), which could reveal the presence of the attack.

In conclusion, the AFI mechanism does not attempt to create an impenetrable defense but rather fundamentally raises the bar for a successful attack. It forces the attacker into a difficult tri-lemma, having to balance attack effectiveness, trigger stealth, and model utility. While the implementation and evaluation of more complex adaptive attacks constitute an important direction for our future work, the above analysis demonstrates that AFI provides a foundation for practical backdoor defense with inherent robustness by making adaptive attacks more costly and difficult to conceal.

5 Conclusion

We propose a feature-comparison fusion strategy that combines the sample under inspection with two contrasting reference samples and leverages model predictions for reverse reasoning, enabling effective identification of poisoned samples. Extensive experiments demonstrate that AFI robustly defends against four common backdoor attacks, outperforming three mainstream defense methods in detection success rate, accuracy, and stability across different datasets and attack types. Overall, AFI provides a novel and practical approach for backdoor attack defense.

Acknowledgement: The authors would like to thank all those who provided support and constructive feedback during the preparation of this work.

Funding Statement: This work was supported by the National Natural Science Foundation of China Grant (No. 61972133), Project of Leading Talents in Science and Technology Innovation for Thousands of People Plan in Henan Province Grant (No. 204200510021), and the Key Research and Development Plan Special Project of Henan Province Grant (No. 241111211400).

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Simin Tang and Zhiyong Zhang; Data curation, Simin Tang; Formal analysis, Simin Tang; Funding acquisition, Zhiyong Zhang; Investigation, Simin Tang; methodology, Simin Tang; Project administration, Zhiyong Zhang; Resources, Simin Tang and Junyan Pan; Software, Simin Tang; Supervision, Zhiyong Zhang, Weiguo Wang and Junchang Jing; Validation,

Simin Tang; Visualization, Simin Tang; Writing—original draft, Simin Tang; Writing—review & editing, Zhiyong Zhang, Junyan Pan, Gaoyuan Quan, Weiguo Wang and Junchang Jing. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets used in this study are publicly available. Specifically, MNIST [31], CIFAR-10 [32], and ImageNet [33] can be accessed from their official repositories. The implementation code is available from the corresponding author upon reasonable request.

Ethics Approval: Not applicable. This study does not involve human participants or animals. The experiments were conducted using publicly available datasets: MNIST [31], CIFAR-10 [32], and ImageNet [33].

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

Abbreviations

AFI	Adaptive Feature Injection
CNN	Convolutional Neural Network
ACC	Accuracy
DSR	Defense Success Rate
DSP	Detection Stability and Portability

References

1. Sze V, Chen YH, Yang TJ, Emer JS. Efficient processing of deep neural networks: a tutorial and survey. Proc IEEE. 2017;105(12):2295–329. doi:10.1109/JPROC.2017.2761740.
2. Gu T, Liu K, Dolan-Gavitt B, Garg S. Evaluating backdooring attacks on deep neural networks. IEEE Access. 2019;7:47230–44. doi:10.1109/ACCESS.2019.2909068.
3. Bai Y, Xing G, Wu H, Rao Z, Ma C, Wang S. Backdoor attack and defense on deep learning: a survey. IEEE Trans Comput Soc Syst. 2024;12(1):404–34. doi:10.1109/TCSS.2024.3482723.
4. Qi F, Yao Y, Xu S, Liu Z, Sun M. Turn the combination lock: learnable textual backdoor attacks via word substitution. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics; 2021 Aug 1–6; Online. p. 4873–83. doi:10.18653/v1/2021.acl-long.377.
5. Zeng Y, Park W, Mao ZM, Jia R. Rethinking the backdoor attacks' triggers: a frequency perspective. In: Proceedings of the IEEE/CVF International Conference on Computer Vision; 2021 Oct 11–17; Montreal, QC, Canada. p. 16473–81. doi:10.1109/ICCV48922.2021.01616.
6. Wang B, Yao Y, Shan S, Li H, Viswanath B, Zheng H. Neural cleanse: identifying and mitigating backdoor attacks in neural networks. In: Proceedings of the IEEE Symposium on Security and Privacy; 2019 May 19–23; San Francisco, CA, USA. p. 707–23. doi:10.1109/SP.2019.00031.
7. Zhu L, Ning R, Wang C, Xin C, Wu H. Gangsweep: Sweep out neural backdoors by GAN. In: Proceedings of the 28th ACM International Conference on Multimedia; 2020 Oct 12–16; Seattle, WA, USA. p. 3173–81. doi:10.1145/3394171.3413546.
8. Bai J, Gao K, Gong D, Xia ST, Li Z, Liu W. Hardly perceptible trojan attack against neural networks with bit flips. In: European Conference on Computer Vision. Cham, Switzerland: Springer; 2022. p. 104–21. doi:10.1007/978-3-031-20065-6_7.
9. Gu T, Dolan-Gavitt B, Garg S. Badnets: identifying vulnerabilities in the machine learning model supply chain. arXiv:1708.06733. 2017.
10. Li Y, Lyu X, Koren N, Lyu L, Li B, Ma X. Anti-backdoor learning: Training clean models on poisoned data. In: Proceedings of the 35th Conference on Neural Information Processing Systems; 2021 Dec 6–14; Online. p. 14900–12.
11. Wenger E, Passananti J, Bhagoji AN, Yao Y, Zheng H, Zhao BY. Backdoor attacks against deep learning systems in the physical world. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021 Jun 19–25; Online. p. 6206–15. doi:10.1109/CVPR46437.2021.00614.

12. Zhao Z, Chen X, Xuan Y, Dong Y, Wang D, Liang K. Defeat: deep hidden feature backdoor attacks by imperceptible perturbation and latent representation constraints. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022 Jun 19–24; New Orleans, LA, USA. p. 15213–22. doi:10.1109/CVPR52688.2022.01478.
13. Nguyen TA, Tran A. Input-aware dynamic backdoor attack. *Adv Neural Inf Process Syst.* 2020;33:3454–64.
14. Liu Y, Ma X, Bailey J, Lu F. Reflection backdoor: a natural backdoor attack on deep neural networks. In: European Conference on Computer Vision. Cham, Switzerland: Springer; 2020. p. 182–99.
15. Nguyen A, Tran A. WaNet: imperceptible warping-based backdoor attack. In: Proceedings of the 9th International Conference on Learning Representations; 2021 May 3–7; Online.
16. Gan L, Li J, Zhang T, Li X, Meng Y, Wu F, et al. Triggerless backdoor attack for NLP tasks with clean labels. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies; 2022 Jul 10–15; Seattle, WA, USA. p. 2942–52.
17. Zhao S, Xu X, Xiao L, Wen J, Tuan LA. Clean-label backdoor attack and defense: an examination of language model vulnerability. *Expert Syst Appl.* 2025;265:125856. doi:10.1016/j.eswa.2024.125856.
18. Chen W, Xu X, Wang X, Zhou H, Li Z, Chen Y. Invisible backdoor attack with attention and steganography. *Comput Vis Image Underst.* 2024;249(1):104208. doi:10.1016/j.cviu.2024.104208.
19. Li Y, Jiang Y, Li Z, Xia ST. Backdoor learning: a survey. *IEEE Trans Neural Netw Learn Syst.* 2024;35(1):5–22. doi:10.1109/TNNLS.2022.3182979.
20. Zhao S, Ma X, Zheng X, Bailey J, Chen J, Jiang YG. Clean-label backdoor attacks on video recognition models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2020 Jun 14–19; Seattle, WA, USA. p. 14443–52. doi:10.1109/CVPR42600.2020.01445.
21. Chen T, Zhang Z, Zhang Y, Chang S, Liu S, Wang Z. Quarantine: sparsity can uncover the trojan attack trigger for free. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022 Jun 19–24; New Orleans, LA, USA. p. 598–609. doi:10.1109/CVPR52688.2022.00068.
22. Guan J, Tu Z, He R, Tao D. Few-shot backdoor defense using Shapley estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022 Jun 19–24; New Orleans, LA, USA. p. 13358–67.
23. Doan K, Lao Y, Zhao W, Li P. LIRA: learnable, imperceptible and robust backdoor attacks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision; 2021 Oct 11–17; Montreal, QC, Canada. p. 11966–76. doi:10.1109/ICCV48922.2021.01175.
24. Salem A, Wen R, Backes M, Ma S, Zhang Y. Dynamic backdoor attacks against machine learning models. In: Proceedings of the 2022 IEEE European Symposium on Security and Privacy (EuroS&P); 2022 Jun 6–10; Genoa, Italy. p. 703–18. doi:10.1109/EuroSP53844.2022.00499.
25. Liu K, Dolan-Gavitt B, Garg S. Fine-pruning: defending against backdooring attacks on deep neural networks. In: International Symposium on Research in Attacks, Intrusions, and Defenses. Cham, Switzerland: Springer; 2018. p. 273–94.
26. Gao Y, Xu C, Wang D, Chen S, Ranasinghe DC, Nepal S. STRIP: a defence against trojan attacks on deep neural networks. In: Proceedings of the Annual Computer Security Applications Conference; 2019 Dec 9–13; San Juan, PR, USA. p. 113–25.
27. Chou E, Tramèr F, Pellegrino G, Boneh D. Sentinel: detecting physical attacks against deep learning systems. *arXiv:1812.00292.* 2018.
28. Rajabi A, Asokraj S, Jiang F, Niu L, Ramasubramanian B, Ritcey J, et al. MDTD: a multi-domain Trojan detector for deep neural networks. In: Proceedings of the ACM Conference on Computer and Communications Security; 2023 Nov 26–30; Copenhagen, Denmark. p. 2232–46.
29. Guo W, Tondi B, Barni M. Universal detection of backdoor attacks via density-based clustering and centroids analysis. *IEEE Trans Inf Forensics Secur.* 2023;19(7):970–84. doi:10.1109/TIFS.2023.3329426.
30. Chen X, Liu C, Li B, Lu K, Song D. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv:1712.05526.* 2017.
31. LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. *Proc IEEE.* 1998;86(11):2278–324. doi:10.1109/5.726791.

32. Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images. In: Technical report. Toronto, ON, Canada: University of Toronto; 2009.
33. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. ImageNet: A large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2009 Jun 20–25; Miami, FL, USA. p. 248–55. doi:10.1109/CVPR.2009.5206848.
34. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA. p. 770–8.
35. Li Y, Lyu X, Koren N, Lyu L, Li B, Ma X. Neural attention distillation: erasing backdoor triggers from deep neural networks. arXiv:2101.05930. 2021.
36. Li Y, Zhai T, Jiang Y, Li Z, Xiao ST. Backdoor attack in the physical world. arXiv:2104.02361. 2021.