Security of Visual Neural Networks: Backdoor Attacks and Adversarial Purification

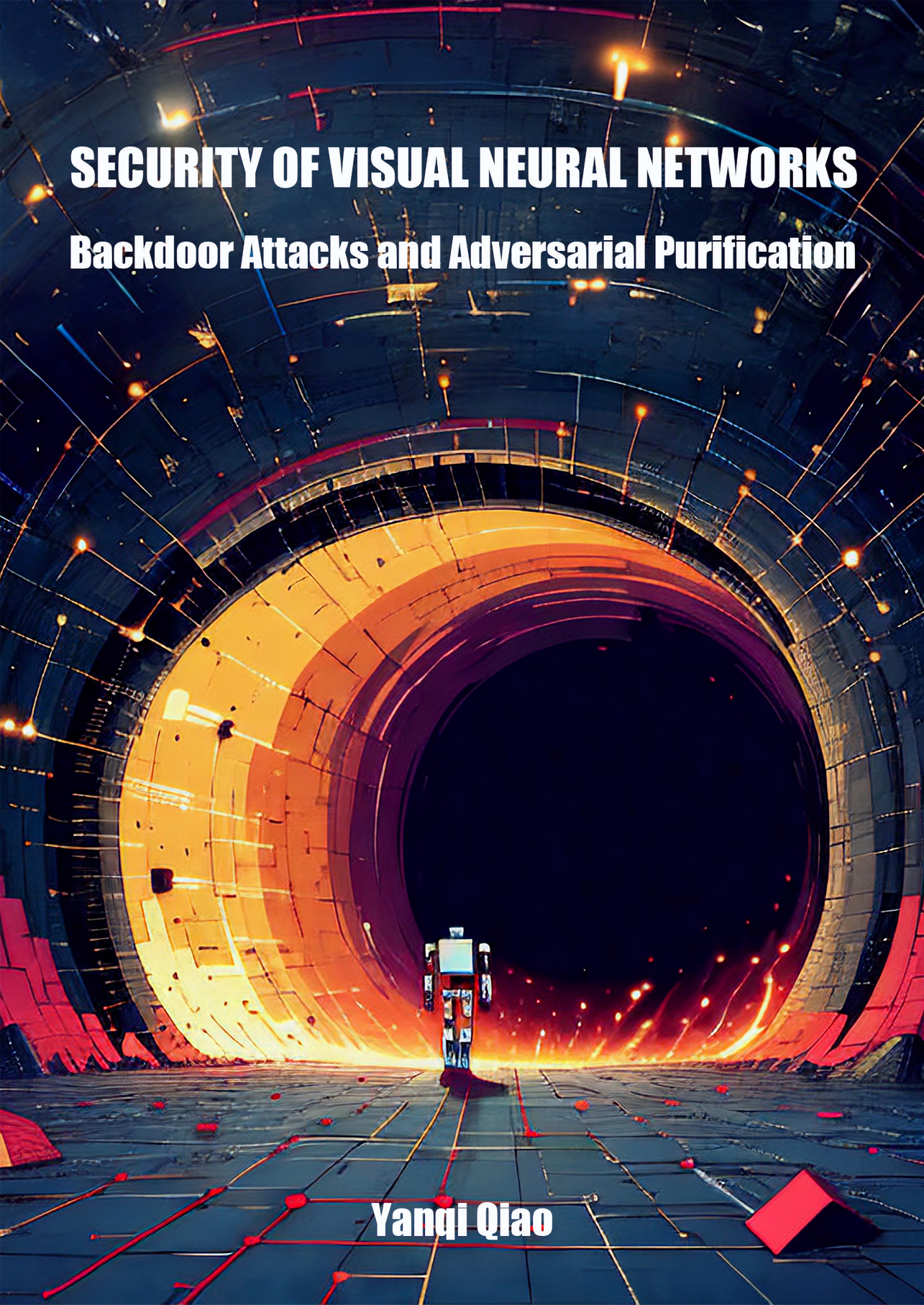
Qiao, Y.

DOI
10.4233/uuid:2e820134-8d68-4acd-a3af-e1dd58c40957

Publication date
2025

Document Version
Final published version

Citation (APA)
Qiao, Y. (2025). *Security of Visual Neural Networks: Backdoor Attacks and Adversarial Purification*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:2e820134-8d68-4acd-a3af-e1dd58c40957


SECURITY OF VISUAL NEURAL NETWORKS
Backdoor Attacks and Adversarial Purification
Yanqi Qiao

Security of Visual Neural Networks: Backdoor Attacks and Adversarial Purification

SECURITY OF VISUAL NEURAL NETWORKS: BACKDOOR ATTACKS AND ADVERSARIAL PURIFICATION

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology
by the authority of the Rector Magnificus, Prof. dr. ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates
to be defended publicly on
Thursday 30 October 2025 at 17.30 o'clock

by

Yanqi QIAO

Master of Science in Cyberspace Security,
Wuhan University, China,
born in Inner Mongolia, China

This dissertation has been approved by the promotors.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Prof.dr.ir. R.L. Lagendijk,	Delft University of Technology, *promotor*
Dr. K. Liang,	Delft University of Technology, *copromotor*
Independent members:	
Prof.dr. C.D. Jensen	Aarhus University, Denmark
Dr. A. Nocera	University of Pavia, Italy
Prof.dr. M.E. van Dijk	VU Amsterdam, NL / CWI, NL
Dr.ir. S. Verwer	Delft University of Technology
Prof.dr. G. Smaragdakis	Delft University of Technology
Prof.dr.ir. M.J.T. Reinders	Delft University of Technology (reserve member)

Keywords: Adversarial Machine Learning, Computer Vision, Federated Learning, Backdoor Attack, Adversarial Purification

Printed by: Ipskamp Printing

Cover by: Dazhuang Liu, Stable Diffusion

ISBN 978-94-6473-956-5

An electronic copy of this dissertation is available at
https://repository.tudelft.nl/.

In memory of my grandfather.

Contents

SUMMARY

Deep learning, a prominent branch of machine learning, leverages artificial neural networks to extract complex patterns and hierarchical representations from large datasets. Notably, advanced architectures such as convolutional neural networks (CNNs) and vision transformers (ViTs) have achieved remarkable success in various computer vision applications, particularly in image classification tasks.

While deep learning offers substantial benefits, it faces security challenges stemming from potentially unreliable models and untrustworthy training data. Such vulnerabilities can compromise model functionality through maliciously perturbed inputs or by introducing model *Trojans*, where adversaries embed triggers in input data to activate harmful behaviors.

Despite significant research on adversarial and backdoor attacks, along with their countermeasures in various deep learning systems, there remains a critical demand for innovative technical solutions to mitigate persistent vulnerabilities and bolster the security and robustness of these systems.

This thesis addresses three key security challenges, including (1) low attack robustness against common image transformations and anomaly frequency perturbations in backdoor triggers under centralized learning; (2) anomaly backdoor features and parameters introduced by current attack methods under decentralized learning; and (3) the significant drop in both clean and robust accuracy caused by global image restoration using diffusion models in adversarial purification.

In examining the vulnerabilities of centralized deep learning systems, Chapter 2 focuses on backdoor attacks against CNNs and Transformers as a malicious data provider. The thesis leverages an evolutionary algorithm to optimize the frequency properties of the designed trigger to maximize attack effectiveness, robustness against image transformation operations, and stealthiness in dual space under the black-box setting.

In investigating the security issues in the decentralized scenarios, Chapters 3 and 4 focuses on backdoor attacks against federated learning from the perspective of a malicious client. In Chapter 3, we propose a backdoor attack to disguise malicious updates of the adversary as benign at the parameter level by backdoor neuron constraint and model camouflage. In Chapter 4, we utilize the power of generative adversarial networks to produce stealthy and flexible triggers that minimize the representation distance between poisoned and benign samples.

To enhance the security of deep learning through data perspective, the thesis focuses on adversarial purification to improve the model robustness against adversarial attacks. In Chapter 5, we identify perturbed image regions through multi-scale superpixel segmentation and occlusion analysis, subsequently using diffusion models for inpainting while maintaining visual consistency.

SAMENVATTING

Deep learning, een prominente tak van machine learning, benut artificial neural networks om complexe patronen en hiërarchische representaties uit grote datasets te halen. Geavanceerde architecturen zoals convolutional neural networks (CNNs) en vision transformers (ViTs) hebben opmerkelijke successen behaald in diverse computer vision-toepassingen, vooral bij taak van beeldclassificatie.

Hoewel deep learning aanzienlijke voordelen biedt, kampt het met veiligheidsuitdagingen door mogelijk onbetrouwbare modellen en trainingsdata. Dit soort kwetsbaarheden kan de modelfunctionaliteit compromitteren via kwaadaardig verstoorde inputs of door *Trojans* in het model, waarbij aanvallers triggers in invoerdata inbedden om schadelijk gedrag te activeren.

Ondanks significant onderzoek naar adversarial en backdoor aanvallen, en hun tegenmaatregelen in diverse deep learning systemen, is er een grote behoefte aan innovatieve technische oplossingen om hardnekkige kwetsbaarheden te verminderen en de veiligheid en robuustheid van deze systemen te versterken.

Deze scriptie behandelt drie belangrijke beveiligingsuitdagingen, waaronder (1) lage aanvalsrobuustheid tegen veelvoorkomende beeldtransformaties en anomaliefrequentieverstoringen in backdoor triggers onder gecentraliseerd leren; (2) anomalie backdoorkenmerken en parameters geïntroduceerd door huidige aanvalsmethoden onder gedecentraliseerd leren; en (3) de significante daling in zowel schone als robuuste nauwkeurigheid veroorzaakt door globale beeldrestauratie met behulp van diffusion models in adversarial purification.

Bij het onderzoeken van de kwetsbaarheden van gecentraliseerde deep learning systemen, richt Hoofdstuk 2 zich op backdoor aanvallen tegen CNNs en ViTs als kwaadaardige data-aanbieder. Het proefschrift benut een evolutionary algorithm om de frequentie-eigenschappen van de ontworpen trigger te optimaliseren voor maximale aanvalseffectiviteit, robuustheid tegen beeldtransformatie-operaties en onopvallendheid in de duale domeinruimte in de black-box omgeving.

Bij het onderzoeken van veiligheidskwesties in gedecentraliseerde scenario's, richten Hoofdstukken 3 en 4 zich op backdoor attacks tegen federated learning vanuit het perspectief van een kwaadaardige klant. In Hoofdstuk 3 introduceren we een backdoor aanval om kwaadaardige updates van de tegenstander als onschuldig te vermommen op parameterniveau door backdoor neuron constraint en model camouflage. In Hoofdstuk 4 gebruiken we de kracht van generative adversarial networks om onopvallende en flexibele triggers te produceren die de representatie-afstand tussen vergiftigde en onschadelijke samples minimaliseren.

Om de veiligheid van deep learning vanuit dataperspectief te verbeteren, richt het proefschrift zich op adversarial purification om de modelrobuustheid tegen adversarial aanvallen te verbeteren. In Hoofdstuk 5 identificeren we verstoorde beeldregio's via

multi-scale superpixel egmentatie en occlusion analyse, waarna we diffusion models gebruiken voor inpainting met behoud van visuele consistentie.

1

INTRODUCTION

Deep Learning is a powerful branch of machine learning that employs deep neural networks (DNNs) to build highly capable models. These models, with their multiple layers of interconnected neurons and extensive training on large datasets, achieve exceptional performance in a wide range of complex real-world tasks. Specifically, DNN architectures such as convolutional neural networks (CNNs) [1–5] and vision transformers (ViTs) [6–8] have achieved significant success in many popular computer vision tasks, e.g., image classification [2, 3, 7, 9, 10], object detection [4, 11, 12] and semantic segmentation [13–16].

Despite their remarkable success, DNN models are vulnerable to various attacks, such as adversarial and backdoor attacks. These vulnerabilities compromise the functionality of DNN models in both centralized and decentralized settings, posing significant risks to safety-critical applications. This motivates the research into enhancing the robustness of DNN models against such attacks. Notable examples of attacks in high-stakes applications include medical image analysis [17], autonomous driving [18, 19], and face recognition systems [20].

While significant research has focused on adversarial machine learning, particularly on adversarial and backdoor attacks, the evolution of these techniques has revealed new security challenges. These challenges demand the evaluation of more sophisticated attack strategies and the proposal of more robust defensive mechanisms to defend against such potential threats. These advancements are crucial for uncovering model vulnerabilities and enhancing their robustness. Among these challenges, (1) low attack effectiveness against common image transformation operations and anomaly frequency perturbations introduced by triggers in centralized backdoor attacks, (2) anomaly backdoor features and parameters that can be easily detected by robust decentralized learning frameworks, and (3) low clean and robust accuracy caused by global image restoration via diffusion models in adversarial purification are considered in the thesis for the following reasons[1]. First, current research lacks a comprehensive exploration of stealthiness in the frequency domain and robustness against common image transformation operations in backdoor

[1] For simplicity, these challenges will be referred to as *(1) robustness and stealthiness in centralized backdoor attacks, (2) stealthiness in decentralized backdoor attacks and (3) robustness in adversarial purification* in the following sections.

attacks. By investigating the stealthiness and robustness of backdoor triggers in the frequency domain, it inspires researchers to inspect potential malicious perturbations in different image domains and develop defensive methods to eliminate the impact of those poisoned samples from a frequency perspective. Second, existing backdooring approaches against federated learning often overlook the anomalies of both backdoor features learned by the victim model and model parameters introduced by backdoor training. Investigating this underexplored area would enable the design of more secure FL aggregation algorithms against stealthy backdoor attacks. Third, current adversarial purification methods exhibit limitations, particularly in defending against color-based attacks and achieving high clean-accuracy due to coarse restoration strategies. Improving the robustness of adversarial purification methods can further improve the robustness of DNN models against adversarial attacks. Collectively, studying more robust and stealthy backdoor attacks in both centralized and decentralized settings helps the machine learning community better understand the vulnerabilities of current DNN models. This knowledge enables the development of stronger defensive methods to safeguard DNNs against such attacks. Additionally, enhancing the robustness of existing AP purification methods not only strengthens the security of DNN models but also provides deeper insights into purifying backdoor triggers. Both attack and defense methods proposed in this thesis aim to offer defensive insights and contributions to the machine learning security community.

In the remainder of this chapter, Section 1.1 provides an overview of popular DNN architectures, federated learning, adversarial attacks, and backdoor attacks. Section 1.2 then discusses the security challenges in both centralized and decentralized machine learning settings, along with the state-of-the-art defenses against adversarial attacks. Then, we present our problem statement and research questions in Section 1.3. Finally, we list the contributions and the outline of the thesis in Section 1.4.

1.1. ADVERSARIAL MACHINE LEARNING

1.1.1. VISUAL NEURAL NETWORKS

Convolutional Neural Networks (CNNs) [1–3, 21, 22] have become effective deep neural networks (DNNs) architectures for various computer vision tasks, e.g., image classification [5, 23], object tracking [24, 25] and detection [26, 27], facial recognition [28]. A typical CNN architecture comprises multiple layers, including convolutional layers, pooling layers and fully-connected (dense) layers. The convolutional layers extract image features by applying convolutional filters that slide across the entire input image. These filters effectively learn to detect features such as edges, textures and patterns, producing feature maps. Then, the feature maps are downsampled by pooling layers (e.g., max pooling) to reduce computational complexity and increase robustness to small spatial shifts. After multiple convolutional and pooling layers, the extracted features are flattened via dense layers to perform final predictions. Activation functions are applied between layers to introduce non-linearity, enabling the network to learn complex, non-linear relationships within the data.

Advanced CNNs have evolved to address diverse challenges, building upon these fundamental components. For example, ResNet [2] incorporates skip connections to address the vanishing gradient problem, easing the training of deeper neural networks. To address the issue of diminishing feature reuse, WideResNet [21] extends ResNet by widening the convolutional layers (i.e., increasing the number of their channels). Xception [29] builds on the idea of depthwise separable convolutions including depthwise convolution and pointwise convolution, which significantly decreased the number of model parameters and computations.

Vision Transformers (ViTs) [30] have revolutionized the field of computer vision, challenging the long-standing dominance of CNNs. ViTs adapt the original transformer arhitecture [31] for natural language processing (NLP) to the domain of computer vision. Given a vision transformer model $\mathscr{F}(\cdot)$ and training dataset $\mathscr{D} = \{(x_i, y_i) | x_i \in \mathbb{R}^{H\times W\times C}, y_i \in \mathbb{R}^K\}_{i=1}^N$, where N is the size of dataset, K is the number of classes, H, W and C are the height, width and channels of an input x, y is the ground-truth label. The input image x is divided into a sequence of $H \times W/p^2$ patches with the shape of $p \times p$. These patches are flattened and linearly projected into embedding vectors, similar to word embeddings in NLP. Moreover, a classification token is added to the head of the above embedding vectors, forming the input token sequence as $T = \{t_{cls}, t_1, t_2, \cdots, t_{H\times W/p^2}\}$. The core component of the ViTs is the self-attention mechanism. Self-attention mechanism allows ViTs to weigh the importance of different patches in relation to each other, effectively capturing long-range dependencies and global context. Each token is used to perform attention map calculation by multi-head self-attention (MSA) module as follows:

$$Attention(T) = Softmax(\frac{TW_Q(TW_K)^T}{\sqrt{d}}TW_V), \tag{1.1}$$

where d is the dimension of the query and the key, W_Q, W_K and W_V are learnable weights of the query Q, key K and value V, respectively. MSA enhances self-attention mechanism by performing it multiple times in parallel with distinct

learned linear projections (heads), allowing ViTs focus on diverse aspects of the input simultaneously. The classification result is derived from multiple MSA attention calculations through a multi-layer perceptron (MLP). Compared to CNNs, ViTs excel at capturing global context and long-range dependencies, offering a weaker inductive bias. In the thesis, we evaluate the performance of backdoor attacks and adversarial purification on such ViT architectures, demonstrating the generalization of the proposed methods to transformer-based models.

1.1.2. FEDERATED LEARNING

Federated Learning (FL) [32–34] is a distributed machine learning framework proposed to preserve data privacy among participating clients. It supports collaborative training of an accurate global model by allowing local clients to share their model updates with a central server without compromising their private datasets. Consider the empirical risk minimization (ERM) in FL setting where the goal is to learn a global classifier $f_\theta : \mathcal{X} \to \mathcal{Y}$ that maps an input $x \in \mathcal{X}$ to a target label $y \in \mathcal{Y}$. Recall that the FL server cannot access the training dataset. It aggregates the parameters/gradients from local agents performing centralized training with local datasets. The de-facto standard rule for aggregating the updates is so-called `FedAvg` [35]. The training task is to learn the global parameters θ by solving the finite-sum optimization:

$$\min_{\theta} f_\theta = \frac{1}{n} \sum_{i=1}^{n} f_{\theta_i}, \tag{1.2}$$

where n is the number of participating agents. At round t, the server S randomly selects $n^t \in \{1, 2, ..., n\}$ agents to participate in the aggregation and send the global model θ^t to them. Each of the agents i trains its local classifier $f_{\theta_i} : \mathcal{X}_i \to \mathcal{Y}_i$ with its local dataset $\mathcal{D}_i = \{(x_j, y_j) : x_j \in \mathcal{X}_i, y_j \in \mathcal{Y}_i, j = 1, 2, ..., N\}$ for some epochs, where $N = |\mathcal{D}_i|$, by certain optimization algorithm, e.g., stochastic gradient descent (SGD). The objective of agent i is to train a local model as:

$$\theta_i^* = \underset{\theta^t}{\operatorname{argmin}} \sum_{(x_j, y_j) \in \mathcal{D}_i} \mathcal{L}(f_{\theta^t}(x_j), y_j), \tag{1.3}$$

where θ^t is the parameters θ of global model at round t, $\mathcal{L}$ stands for the classification loss, e.g., cross-entropy loss. Then agent i computes its local update as $\delta_i^t = \theta_i^* - \theta^t$, and sends back to S. Finally, the server aggregates all updates and produces the new global model with an average as follows:

$$\theta^{t+1} = \theta^t + \frac{\gamma}{|n^t|} \sum_{i \in n^t} \delta_i^t, \tag{1.4}$$

where γ is the global learning rate. When the global model θ converges or the training reaches a specific iteration upper bound, the aggregation process terminates and outputs a final global model. During inference, given a benign sample x and its true label y, the learned global classifier f_θ will behave well as: $f_\theta(x) = y$.

Optimizations of FL have been proposed for various purposes, e.g., privacy [36], security [37, 38], heterogeneity [39], communication efficiency [40, 41] and personalization issues [42, 43].

1.1.3. ADVERSARIAL ATTACKS AND BACKDOOR ATTACKS

Adversarial attacks aim to generate adversarial examples with only minimal perturbation based on test samples whose classification results can be changed to an attacker-specified class during deployment. We focus on adversarial attacks against deep neural networks, denoted by $f_\theta(x)$, where θ represents the model's parameters and $x \in \mathscr{X}$ is the input data, belonging to the input space $\mathscr{X} \subset \mathbb{R}^d$. The output of the DNN is a prediction vector $f_\theta(x) \in \mathbb{R}^K$, where K is the number of classes. We assume a classification setting where the predicted class is given by $\arg\max_k f_\theta(x)_k$.

An adversarial example is produced by a carefully crafted perturbation δ based on a benign input x, resulting in $x' = x + \delta$. This perturbation is designed to be imperceptible to humans, thus constrained in magnitude, $|\delta|_p \leq \epsilon$, while simultaneously causing the DNN f_θ to misclassify the input into target label. Formally, an adversarial example x' satisfies:

$$f_\theta(x') = t \neq f_\theta(x) \quad \text{(Misclassification)}, \tag{1.5}$$

$$\|\delta\|_p < \epsilon \quad \text{(Perceptual similarity)}, \tag{1.6}$$

where $|\cdot|_p$ denotes the ℓ_p-norm, commonly ℓ_∞ (maximum perturbation per feature), ℓ_2 (Euclidean distance), or ℓ_1 (Manhattan distance), and ϵ represents the maximum permissible perturbation magnitude (the perturbation budget). The proportion of adversarial examples correctly predicted as t is known as the attack success rate (ASR). Effectively, generating an adversarial example involves maximizing the loss function of the classifier, f_θ (or similarly, minimizing the classifier's confidence in the true label):

$$\delta = \arg\max_{||\delta|| \leq \epsilon} \mathscr{L}(f_\theta(x+\delta), y), \tag{1.7}$$

where $\mathscr{L}$ is the loss function and y is the true label of x.

Based on various attack capabilities and scenarios, existing attacks can be roughly divided into three categories: white-box and black-box adversarial attacks, transferable attacks. White-box attacks include optimization-based methods [44–49] that generates adversarial examples by minimizing the distances from the original samples, and universal adversarial attacks [50] that designed to create small universal perturbations capable of inducing misclassification across most images. Without any prior knowledge of the model architecture or training data, black-box attacks can only interact with a pre-trained ML model by querying it on various data samples and obtaining the model's confidence scores [51–55] or its prediction [56–59]. Other studies [60–64] explore the transferability of adversarial attacks, where an attacker performs white-box adversarial attacks on a pre-trained substitute model and transfers the attacks to a target model.

Backdoor attacks involve manipulating clean data by embedding a *trigger* and altering the label to a target label, resulting in a poisoned dataset. An adversary, as a model provider, can supply users with a poisoned model trained on this dataset. During inference, the poisoned model produces the adversary's desired output when the test input contains a trigger while it functions normally on clean input.

Considering backdoor attacks on image classification, Let $f_\theta : \mathscr{I} \to \mathbb{R}^K$ be an image classifier parameterized with θ that maps an input image $\mathscr{I} \subseteq [0,1]^{H \times W \times C}$ to an

output class, where K is the number of classes, H, W and C are the height, width and channels of an input image. The parameters θ of the classifier are learned using a training dataset $\mathscr{D}_c = \{(x_i, y_i)|x_i \in \mathscr{I}, y_i \in \mathbb{R}^K\}_{i=1}^{N}$. In a standard backdoor attack, the attacker crafts a subset of $\mathscr{D}_c$ with ratio ρ to produce the poisoned dataset $\mathscr{D}_p = \{(x_i', y_i')|x_i' \in \mathscr{I}, y_i' \in \mathbb{R}^K\}_{i=1}^{N\times\rho}$ by the trigger function $\mathscr{T}(\cdot)$ and target label function $\eta(\cdot)$. Given a clean image x from the clean subset and its true class y, the commonly used $\mathscr{T}(\cdot)$ and $\eta(\cdot)$ are defined with a hyper-parameter $m \in [0,1]$ and a trigger pattern t as follows:

$$x' = \mathscr{T}(x, m, t) = x \cdot (1 - m) + t \cdot m, \quad y' = \eta(y) = y_t, \tag{1.8}$$

where y_t is the target class. Under empirical risk minimization, a typical attack aims to inject backdoors into the classifier f by learning θ with both clean dataset $\mathscr{D}_c$ and poisoned dataset D_p so that the classifier misclassifies the poisoned data into the target class while behaving normally on clean data. The optimization problem is defined as follows:

$$\min_{\theta} \sum_{(x,y)\in\mathscr{D}_c} \mathscr{L}(f_\theta(x), y) + \sum_{(x',y')\in\mathscr{D}_p} \mathscr{L}(f_\theta(x'), y'), \tag{1.9}$$

where $\mathscr{L}$ represents the cross-entropy loss.

[65] introduces the first backdoor attack against deep learning models. It employs a patch-based pattern as the trigger, injecting it into a small fraction of clean data, which causes the victim model to misclassify those poisoned images to the target label. To enhance the invisibility of triggers for bypassing human inspection, some works [66–71] focus on imperceptible backdoor attacks in the pixel domain. Additionally, recent works [17, 72–76] explore the frequency domain of the input image to naturally guarantee invisibility due to frequency properties. Later, several works [77–80] reveal the importance of stealthiness in latent feature space. There are other types of attacks tailored to different scenarios such as clean-label attacks [81–84], where poisoned inputs and their labels appear consistent to a human.

1.2. SECURITY CHALLENGES IN VISUAL NEURAL NETWORKS

Building upon the foundational concepts of visual neural networks, adversarial machine learning and decentralized settings, this section presents a comprehensive review of state-of-the-art attack and defensive mechanisms. The discussion emphasizes their inherent limitations and challenges, critically examining these issues from both adversarial and defensive perspectives. Key challenges identified include: (1) robustness and stealthiness in centralized backdoor attacks; (2) stealthiness in decentralized backdoor attacks, and (3) robustness in adversarial purification methods. By addressing the first challenge, our work introduces a more stealthy and robust frequency backdoor in black-box scenarios. This serves as a critical warning for follow-up defense research, underscoring the need for closer scrutiny of minimal malicious manipulations in the frequency domain of input images. By addressing the second challenge, we propose two stealthy backdoor attacks against FL that bypass most existing defenses. These attacks expose a

critical weakness in current robust FL aggregation algorithms, urging researchers to develop fine-grained detection mechanisms for local model updates to mitigate their harmful effects. By addressing the third challenge, we propose a more robust AP method that enables targeted removal of potential perturbations to defend against adversarial attacks. This provides a more fine-grained purification strategy for defending DNNs against adversarial attacks, significantly enhancing the robustness of existing AP methods. At the same time, this approach also offers insights for defending against black-box backdoor attacks. In summary, addressing the first two challenges can directly reveal potential attack risks in current deep learning models while also inspiring further research into robust backdoor defenses. Solving the final challenge, however, will not only enhance the robustness of existing deep learning models but also establish a foundation for future defense research. It is important to acknowledge that while this study does not directly address certain challenges in the fields such as membership inference attacks (MIAs), this omission should not be interpreted as diminishing their significance or relevance to the broader field of adversarial machine learning. This is because survey papers in the field of adversarial machine learning, such as [85], typically highlight backdoor attacks and membership inference attacks as distinct key problems (security and privacy). It is explicitly stated that research on backdoor attacks itself has already become a highly specialized subfield (against various scenarios, computer vision tasks, and models), while MIAs utilize different technical approaches to predict whether a data sample belongs to the models training set or not.

ROBUSTNESS AND STEALTHINESS IN CENTRALIZED BACKDOOR ATTACKS

Prior backdoor attacks [65, 66, 70] inject triggers into the pixel domain of image. Since spatial domain contains abundant semantic information, manipulating triggers in image pixels can be easily detected by human inspection. Works [67–70, 78] improve the natural stealthiness by designing imperceptible triggers into the spatial domain. While these methods achieve practical natural stealthiness, they introduce abnormal feature representations in triggers that are distinct from the benign features of clean samples compared to the benign features of clean samples and thus can be detectable by state-of-the-art backdoor defenses. Assuming a white-box setting where the attackers have full control of data, training process and models, works [68, 77–79] modify the training process and objective functions to achieve superior stealthiness in both input and feature representation space, bypassing both most backdoor defenses and explanation tools [86]. However, this assumption is not practical in many real-world applications where the attacker has no prior knowledge of victim model and control of training process. Recent works show that backdoor attacks can inject trigger patterns into frequency space [17, 72, 73, 75]. However, those attacks perturbs the components in the high-frequency region and thus trigger robustness can be harmed by common image transformation operations such as lowpass filters. Moreover, both spatial and frequency triggers introduce distinguishable artifacts when transformed to frequency space.

Inspired by [71] and the above description, an ideal and practical backdoor attack should achieve four objectives, namely, *functionality preservation*, *effectiveness*,

dual-space stealthiness, and *robustness*. Finding such a frequency trigger in a black-box scenario is not trivial. Due to the absence of the victim model and training process, one may handcraft the frequency triggers as in [17, 73, 76] but it could lead to improper frequency properties of the trigger. For example, a large perturbation can disrupt invisibility of poisoned images, while a small perturbation could hinder the model's ability to learn features of a trigger, resulting in low attack effectiveness. Furthermore, selecting an improper frequency band for trigger insertion can compromise the attack effectiveness and robustness.

STEALTHINESS IN DECENTRALIZED BACKDOOR ATTACKS

Similar to the centralized settings, FL is susceptible to backdoor attacks [87–95]. Current backdoor attacks against FL can poison data and models. In data poisoning [89, 96], the attacker poisons the benign samples with a trigger pattern and marks them as a target label in order to induce the model to misbehave by training this poisoned dataset. As for model poisoning [90, 91, 94], the attacker manipulates the training process to achieve their attack objectives such as durability and effectiveness. [78] claims that backdoor triggers can be designed to be imperceptible in the input space [69, 70, 97] and in the latent space [79], or to possess multi-target payloads [98]; the ground-truth labels of the poisoned samples can also align with the intended target label [81–83]. All characteristics contribute to the stealthiness of backdoor attacks. In this thesis, we investigate stealthiness in decentralized backdoor attacks across three key aspects: (1) the imperceptibility in the input space, (2) the imperceptibility in the latent space, and (3) the ability to bypass defenses. In centralized settings, malicious manipulation of training data can be directly detected and mitigated [99]. However, in decentralized settings, where local data is inaccessible, defenders can only use differences in local model updates between malicious and benign clients to identify potential threats. Consequently, evading FL detection does not require achieving imperceptibility in the input space. Nevertheless, we still consider input-space imperceptibility for two reasons. First, like centralized backdoor attacks, a visible trigger [89, 90, 92, 94] can be easily detected by human and machine inspection during global model inference. Second, visible triggers introduce large perturbations, resulting in more distinguished backdoor features. This could be easily reversed and detected by state-of-the-art defenses such as [38, 100][2].

To fully exploit the distributed attributes of FL, [89] only uses parts of the global trigger to generate poisoned images in each malicious client, while the ultimate adversarial goal is still the same as centralized attack – using the global trigger to attack the global model. Moreover, [91] leverages the data from the tail of the input data distribution as poisoned samples. Existing attack methods against FL failed to achieve stealthiness at the model update level (or in the latent space) since they do not provide enough "stealthiness" of the hidden features of the poisoned samples. These features of trigger patterns extracted from convolutional filters *standalone* compared to the benign counterparts during backdoor training, causing

[2]The empirical studies in Chapter 2 also demonstrate that invisible triggers exhibit greater resilience against trigger inversion mechanisms.

distinct weight outliers. Additionally, in fully-connected layers, the backdoor training is to establish a new routing [101, 102], separated from benign ones, between the independent hidden features of attacker's trigger and its corresponding target label, which yields an anomaly at the parameter level. The cause of this anomaly is natural, since the output neurons for the target label must contribute to both benign and backdoor routing, which requires significant weight/bias adjustments to the neurons involved. Therefore, the current defenses [37, 103–108] can easily detect malicious updates of these attacks by exploiting the distinguishable dissimilarity between model (parameter) updates from malicious and benign agents.

ROBUSTNESS IN ADVERSARIAL PURIFICATION

Adversarial attacks [109–112] introduce imperceptible image perturbations, leading to misclassifications and significant safety concerns [113, 114]. To enhance the robustness of DNNs, works [113, 115, 116] introduce adversarial training (AT) mechanism that train models using adversarial examples. While effective against known attacks, AT suffers from overfitting and struggles with unseen attacks and image corruptions [117–119]. Moreover, AT often degrades standard accuracy while increasing computational complexity [120]. Other works [121–123] offer an alternative technique, i.e., adversarial purification (AP), that performs images preprocessing to remove adversarial perturbations before inference However, AP generally shows lower standard accuracy than AT [124, 125]. Creating effective purification models, especially for large-scale datasets, remains challenging due to the inherent trade-off between preserving image semantics and removing perturbations [126]. Thanks to the power of diffusion models, generative AP-based works [127, 128] provides superior image quality, exceeding even GANs in image generation [129, 130], make them well-suited. The inherent denoising process aligns with purification, and their stochastic nature offers potential for robust stochastic defenses [131]. These properties make diffusion models a compelling area for improving DNN robustness against adversarial attacks.

Despite significant progress in defending against adversarial attacks, current generative AP methods face three major challenges. Firstly, existing AP techniques [126, 132, 133] are vulnerable to color-based attacks. This vulnerability originate from the inherent sensitivity of Denoising Diffusion Probabilistic Models (DDPMs) to image colors [132], making them susceptible to even subtle chromatic manipulations introduced by adversaries. Secondly, AP generally shows lower standard accuracy than AT [124, 125]. Creating effective purification models, especially for large-scale datasets, remains challenging due to the inherent trade-off between preserving image semantics and removing perturbations [126]. Thirdly, current AP methods often struggle with defending unseen attacks [118, 119, 134].

1.3. PROBLEM STATEMENT

Adversarial machine learning, including both attacks and corresponding counter-measures across diverse contexts, has emerged as a significant research field within machine learning and security. The thesis addresses critical challenges in enhancing

the robustness and stealthiness of data in centralized backdoor attacks, improving the stealthiness of decentralized backdoor attacks, and developing robust defensive mechanisms against adversarial attacks. As discussed in Section 1.2, while addressing the first two challenges involves proposing stronger attacks against current DNN models and their defenses, our deeper goal is to uncover potential attack risks in existing models. This insight aims to inspire follow-up researchers to design more robust defenses against such threats. To strengthen the robustness of current DNN models in the centralized settings, we tackle the final challenge while also deriving insights to defend against the method introduced in solving the first challenge. Here, we propose the research questions of the thesis, which can be explained in two parts: enhancing stealthiness and robustness in backdoor attacks and improving the robustness of adversarial purification.

STEALTHINESS AND ROBUSTNESS IN BACKDOOR ATTACKS

Stealthiness in centralized backdoor attacks refers to the imperceptible perturbations of trigger patterns within both spatial and frequency domains while robustness denotes the ability of poisoned images to withstand common transformation operations. Most attack methods focus on the natural stealthiness of triggers but relatively few mechanisms investigate the potential for injecting triggers into the frequency domain. Existing frequency-based attacks fail to optimize triggers for both stealthiness in the frequency domain and robustness under practical attack scenarios, which brings the following question:

Q1: *How can effectiveness, dual-domain stealthiness, and robustness be simultaneously achieved in backdoor attacks under practical settings?*

By answering **Q1**, we reveal a critical security threat in real-world scenarios: an attacker could release poisoned datasets with imperceptible frequency triggers that remain robust against common image transformations. Such an attack can have severe consequences in high-stakes applications, particularly because training high-performing models requires vast amounts of data, often leading to the collection of publicly available but harmful datasets. More importantly, exploring this question could advance research on detecting malicious and minimal perturbations in the frequency domain. This would ultimately improve the ability of backdoor defenses to identify and eliminate such stealthy and robust triggers.

In decentralized deep learning systems, stealthiness in backdoor attacks refers to an adversary's capability to send stealthy model updates to the server thereby circumventing robust FL detection mechanisms. Current attack methods often introduce anomalous parameters in their malicious model updates, rendering them detectable by existing defenses. Additionally, these triggers are often perceptible under human inspection. This brings the following question:

Q2: *How can malicious updates be disguised as benign ones at the parameter level to bypass current detection strategies while still maintaining the effectiveness of backdoor attacks?*

A closer investigation of local backdoor training by adversaries reveals that current methods often inject universal trigger patterns into poisoned samples which introduce anomalous features, distinguishing them from the benign features of benign counterparts. This distinction serves as the primary cause of the distinguishable differences between malicious and benign model updates. Consequently, the following question arises:

Q3: *How to eliminate the anomalies introduced during backdoor training while making the trigger sufficiently stealthy for inference under FL settings?*

By answering **Q2** and **Q3**, we uncover a severe security vulnerability in federated learning: malicious clients can disguise themselves as benign ones and submit harmful updates to poison the global model. Simply analyzing the similarity of model updates without deeper statistical information about those updates is insufficient to detect stealthy backdoor attacks in FL. Consequently, exploring these questions inspires follow-up research to detect indistinguishable malicious model updates with more sophisticated defensive mechanisms.

ROBUSTNESS IN ADVERSARIAL PURIFICATION

Robustness in adversarial purification is the ability of deep learning models to maintain robust accuracy when processing adversarial samples through AP techniques. Current AP-based mechanisms typically apply image restoration operations globally, which can destroy features of clean images, thereby reducing clean accuracy after purification. Moreover, these methods are vulnerable to color-based attacks due to the inherent sensitivity of DDPM to image color variations. These limitations collectively impair both the robust accuracy against adversarial samples and the standard accuracy on clean images, which brings the following question:

Q4: *How can adversarial perturbations be effectively mitigated while preserving fine-grained clean features to maintain high clean accuracy during adversarial purification?*

By answering **Q4**, we enhance the robustness of current AP methods against adversarial attacks in a fine-grained way. Additionally, exploring this question also inspires follow-up research on restoring the target malicious region using advanced diffusion models, which can preserve clean features to maintain high clean accuracy.

1.4. CONTRIBUTION OF THE THESIS

The thesis consists of 6 chapters, with each technical chapter (Chapters 2~5) serving as a self-contained replication of a research paper. Each technical chapter is designed to be independently comprehensible, allowing standalone reading and understanding. While efforts have been made to retain the technical rigor and details of the original publications, minor modifications are implemented to ensure coherence within the thesis. Consequently, readers might encounter variations in

notations, overlapping introductory sections, and recurring literature reviews across chapters. The organization of the thesis is outlined as follows:

CHAPTER 2
BACKDOOR ATTACK AGAINST DEEP NEURAL NETWORKS

This chapter addresses the research question **Q1**. Specifically, we first introduce a novel perspective of frequency trigger design that provides dual-space stealthiness and robustness against image transformations, and develop a practical low-frequency backdoor attack method. To achieve this, we formulate the problem as a constrained optimization task to search for optimal frequency triggers that meet all attack objectives. To solve this optimization problem effectively, we employ an evolutionary algorithm, i.e., simulated annealing, which systematically minimizes the attack objectives. Extensive experiments empirically demonstrate that our proposed method achieves state-of-the-art effectiveness and exhibits superior dual-domain stealthiness. This chapter shows that existing mainstream backdoor defenses as well as common image preprocessing techniques are not effective enough to mitigate the attack. Moreover, this chapter aims to inspire the follow-up studies to defend against such stealthy frequency attacks. The chapter closely follows the content of the paper titled *"Low-Frequency Black-Box Backdoor Attack via Evolutionary Algorithm"* by Qiao, Y., Liu, D., Wang, R., Liang, K., in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2025, with minor extensions.

CHAPTER 3
STEALTHY BACKDOOR ATTACK AGAINST FEDERATED LEARNING

This chapter addresses the research question **Q2**. We first design a frequency trigger injection function to produce imperceptible poisoned samples and thus achieve the stealthiness in the input space. We analyze the behavior of neurons in backdoor training at the parameter level by using a task-sensitive neuron searcher (TNS) to identify those backdoor neurons that significantly contribute to backdoor tasks and reduce the impact of backdoor-sensitive neurons. To constrain the impact of backdoor neurons and align model parameters with benign ones, we apply a step-forward training approach to generate benign and malicious models. Furthermore, we apply TNS for the malicious model to find the list of backdoor neurons and minimize its impact in order to evade anomaly parameter detection; meanwhile, we use the benign model as an estimation of the attacker's expected local model. We restrain parameter dissimilarity to make malicious updates indistinguishable from benign updates trained by the attacker without sacrificing the utility of the global model. Therefore, we fully take advantage of the attacker's ability to provide the criterion of malicious model update direction. Finally, we evaluate the attack performance and stealthiness against the most recent robust aggregation algorithms on real-world datasets with various datasets and models. This chapter shows that existing robust FL aggregation rules are not effective enough to mitigate such stealthy attacks. This requires follow-up defensive methods that not only compare parameter similarities across model updates but also invert local updates to

analyze corresponding dummy datasets, helping identify potential malicious clients. In conclusion, this chapter calls for researchers to develop novel defensive strategies that can effectively identify and filter out malicious updates disguised as benign ones. The chapter closely follows the content of the paper titled *"Stealthy Backdoor Attack against Federated Learning through Frequency Domain by Backdoor Neuron Constraint and Model Camouflage"* by Qiao, Y., Liu, D., Wang, R., Liang, K., in IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS), 2024, with minor extensions.

CHAPTER 4

GENERATIVE BACKDOOR ATTACK AGAINST FEDERATED LEARNING

This chapter addresses the research question **Q3**. We propose a stealthy generator-assisted backdoor attack tailored for robust FL which produces naturally imperceptible triggers during the inference stage. These triggers provide hidden feature similarity of benign data and effectively guide poisoned data to reuse benign routing paths associated with the target label. Hereby our approach circumvents parameter anomalies typically observed in malicious updates. We propose a bi-level and constrained optimization problem to find our optimal generator in each iteration efficiently and a customized learning process to solve it with reasonable complexity. Finally, we present intensive experiments to empirically demonstrate that the proposed attack provides state-of-the-art effectiveness and stealthiness against existing defense mechanisms. This chapter shows that most FL defenses are not capable of mitigating such adaptive attacks. This demands more sophisticated defensive strategies in the future, requiring fine-grained inspection of all model updates in each round or even reverse-engineering model updates to acquire statistical information of local training data. Finally, this chapter seeks to motivate future research on new defenses against such adaptive poisoning attacks on FL. The chapter closely follows the content of the paper titled *"FTA: Stealthy Backdoor Attack with Flexible Triggers against Federated Learning"* by Qiao, Y., Liu, D., Panaousis, M., Conti, M., and Liang, K., which is under review from IEEE Transactions on Artificial Intelligence (2025), with minor extensions.

CHAPTER 5

DIFFUSION-BASED PURIFICATION AGAINST ADVERSARIAL ATTACKS

This chapter addresses the research question **Q4**. We, for the first time, apply the denoising diffusion probabilistic model (DDPM) inpainting technique for adversarial purification (AP), enabling targeted removal of perturbations while preserving intrinsic features of clean images, unlike conventional diffusion models that typically operate on a full image scale. Then, we propose leveraging occlusion sensitivity maps to identify regions potentially manipulated by adversarial attacks. This targeted strategy facilitates the precise localization and removal of adversarial perturbations, leading to improved robustness compared to methods relying on non-targeted image restoration. Empirical evaluations demonstrate the superior robustness of our approach in defending color-based attacks and unseen attacks compared to

existing AP techniques. In conclusion, this chapter enhances the robustness of DNN models against adversarial attacks. Furthermore, this chapter provides a potential defensive insight for black-box data poisoning attacks under the centralized settings as in Chapter 2. The chapter closely follows the content of the paper titled *"MSID: Multi-Scale Diffusion-Based Inpainting Defense Against Adversarial Attacks"* by Popovici, A., Qiao, Y., Liu, D., Smaragdakis, G. and Liang, K., which is under review, with minor extensions.

1.4.1. LIST OF EXCLUDED PUBLICATIONS

In the following, we list the papers published during Ph.D. but not included in the thesis since they only present partial elements of the chapters.

- Liu, D.*, **Qiao, Y.***, Wang, R., Liang, K., Smaragdakis, G., 2025. LADDER: Multi-objective Backdoor Attack via Evolutionary Algorithm. In the Network and Distributed System Security (NDSS) Symposium.
- Liu, D., **Qiao, Y.,** Wang, R., Liang, K., Smaragdakis, G., 2025. PASTA: A Patch-Agnostic Twofold-Stealthy Backdoor Attack on Vision Transformers. Under review.
- Stenhuis, R., Liu, D., **Qiao, Y.**, Conti, M., Panaousis, M. and Liang, K., 2025. MeetSafe: Enhancing Robustness against White-box Adversarial Examples. In Frontiers in Computer Science.
- Amalan, A., Wang, R., **Qiao, Y.**, Panaousis, E. and Liang, K., 2022. MULTI-FLGANs: Multi-Distributed Adversarial Networks for Non-IID distribution. arXiv preprint arXiv:2206.12178.
- Tian, Y., Wang, R., **Qiao, Y.**, Panaousis, E. and Liang, K., 2023. FLVoogd: Robust And Privacy Preserving Federated Learning. In Asian Conference on Machine Learning (ACML).

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner. 'Gradient-based learning applied to document recognition'. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[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] K. Simonyan and A. Zisserman. 'Very Deep Convolutional Networks for Large-Scale Image Recognition'. In: *International Conference on Learning Representations*. 2015.

[4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich. *Going Deeper with Convolutions*. 2014. arXiv: 1409.4842 [cs.CV].

[5] A. Krizhevsky, I. Sutskever and G. E. Hinton. 'ImageNet Classification with Deep Convolutional Neural Networks'. In: *Advances in Neural Information Processing Systems* 25 (2012).

[6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.* 'An image is worth 16x16 words: Transformers for image recognition at scale'. In: *arXiv preprint arXiv:2010.11929* (2020).

[7] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin and B. Guo. 'Swin transformer: Hierarchical vision transformer using shifted windows'. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 10012–10022.

[8] X. Dong, J. Bao, D. Chen, W. Zhang, N. Yu, L. Yuan, D. Chen and B. Guo. 'Cswin transformer: A general vision transformer backbone with cross-shaped windows'. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 12124–12134.

[9] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles and H. Jégou. 'Training data-efficient image transformers & distillation through attention'. In: *International conference on machine learning*. PMLR. 2021, pp. 10347–10357.

[10] K. Yuan, S. Guo, Z. Liu, A. Zhou, F. Yu and W. Wu. 'Incorporating convolution designs into visual transformers'. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 579–588.

[11] J. Beal, E. Kim, E. Tzeng, D. H. Park, A. Zhai and D. Kislyuk. 'Toward transformer-based object detection'. In: *arXiv preprint arXiv:2012.09958* (2020).

[12] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov and S. Zagoruyko. 'End-to-end object detection with transformers'. In: *European conference on computer vision*. Springer. 2020, pp. 213–229.

[13] J. Long, E. Shelhamer and T. Darrell. 'Fully convolutional networks for semantic segmentation'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3431–3440.

[14] H. Noh, S. Hong and B. Han. 'Learning deconvolution network for semantic segmentation'. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1520–1528.

[15] R. Strudel, R. Garcia, I. Laptev and C. Schmid. 'Segmenter: Transformer for semantic segmentation'. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 7262–7272.

[16] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez and P. Luo. 'SegFormer: Simple and efficient design for semantic segmentation with transformers'. In: *Advances in neural information processing systems* 34 (2021), pp. 12077–12090.

[17] Y. Feng, B. Ma, J. Zhang, S. Zhao, Y. Xia and D. Tao. 'Fiba: Frequency-injection based backdoor attack in medical image analysis'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 20876–20885.

[18] Z. Ni, R. Ye, Y. Wei, Z. Xiang, Y. Wang and S. Chen. 'Physical backdoor attack can jeopardize driving with vision-large-language models'. In: *arXiv preprint arXiv:2404.12916* (2024).

[19] X. Han, G. Xu, Y. Zhou, X. Yang, J. Li and T. Zhang. 'Physical Backdoor Attacks to Lane Detection Systems in Autonomous Driving'. In: *Proceedings of the 30th ACM International Conference on Multimedia*. New York, NY, USA: Association for Computing Machinery, 2022, pp. 2957–2968.

[20] M. Xue, C. He, J. Wang and W. Liu. 'Backdoors hidden in facial features: a novel invisible backdoor attack against face recognition systems'. In: *Peer-to-Peer Networking and Applications* 14 (2021), pp. 1458–1474.

[21] S. Zagoruyko and N. Komodakis. 'Wide Residual Networks'. In: *British Machine Vision Conference 2016, BMVC 2016* 2016-September (May 2016), pp. 1–87.

[22] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. C. Chen. 'MobileNetV2: Inverted Residuals and Linear Bottlenecks'. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2018), pp. 4510–4520.

[23] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard and L. Jackel. 'Handwritten digit recognition with a back-propagation network'. In: *Advances in neural information processing systems* 2 (1989).

[24] Y. Wu, J. Lim and M.-H. Yang. 'Online Object Tracking: A Benchmark'. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013.

[25] L. Zheng, M. Tang, Y. Chen, G. Zhu, J. Wang and H. Lu. 'Improving Multiple Object Tracking With Single Object Tracking'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 2453–2462.

[26] Z.-Q. Zhao, P. Zheng, S.-t. Xu and X. Wu. 'Object detection with deep learning: A review'. In: *IEEE transactions on neural networks and learning systems* 30.11 (2019), pp. 3212–3232.

[27] Z. Zou, K. Chen, Z. Shi, Y. Guo and J. Ye. 'Object detection in 20 years: A survey'. In: *Proceedings of the IEEE* 111.3 (2023), pp. 257–276.

[28] M. Wang and W. Deng. 'Deep face recognition: A survey'. In: *Neurocomputing* 429 (2021), pp. 215–244.

[29] F. Chollet. 'Xception: Deep learning with depthwise separable convolutions'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.

[30] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit and N. Houlsby. 'An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale'. In: *International Conference on Learning Representations* (2020).

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez,. Kaiser and I. Polosukhin. 'Attention Is All You Need'. In: *Advances in Neural Information Processing Systems* 2017-December (2017), pp. 5999–6009.

[32] H. Yu, Q. Yang, Y. Kang, Y. Cheng, Y. Liu and T. Chen. *Federated learning*. Morgan & Claypool Publishers, 2019.

[33] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen and H. Yu. 'Horizontal federated learning'. In: *Federated learning*. Springer, 2020, pp. 49–67.

[34] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen and H. Yu. 'Vertical federated learning'. In: *Federated Learning*. Springer, 2020, pp. 69–81.

[35] B. McMahan, E. Moore, D. Ramage, S. Hampson and B. A. y Arcas. 'Communication-efficient learning of deep networks from decentralized data'. In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 1273–1282.

[36] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. McMahan, S. Patel, D. Ramage, A. Segal and K. Seth. 'Practical secure aggregation for federated learning on user-held data. arXiv 2016'. In: *arXiv preprint arXiv:1611.04482* 13 ().

[37] P. Blanchard, E. M. El Mhamdi, R. Guerraoui and J. Stainer. 'Machine learning with adversaries: Byzantine tolerant gradient descent'. In: *Advances in Neural Information Processing Systems* 30 (2017).

[38] B. Zhao, P. Sun, T. Wang and K. Jiang. 'Fedinv: Byzantine-robust federated learning by inversing local model updates'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 8. 2022, pp. 9171–9179.

[39] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar and V. Smith. 'Federated optimization in heterogeneous networks'. In: *Proceedings of Machine learning and systems* 2 (2020), pp. 429–450.

[40] Y. Liu, Y. Kang, X. Zhang, L. Li, Y. Cheng, T. Chen, M. Hong and Q. Yang. 'A communication efficient collaborative learning framework for distributed features'. In: *arXiv preprint arXiv:1912.11187* (2019).

[41] J. Jiang, S. Ji and G. Long. 'Decentralized knowledge acquisition for mobile internet applications'. In: *World Wide Web* 23.5 (2020), pp. 2653–2669.

[42] T. Li, S. Hu, A. Beirami and V. Smith. 'Ditto: Fair and robust federated learning through personalization'. In: *International conference on machine learning*. PMLR. 2021, pp. 6357–6368.

[43] T. Yu, E. Bagdasaryan and V. Shmatikov. 'Salvaging federated learning by local adaptation'. In: *arXiv preprint arXiv:2002.04758* (2020).

[44] C. Szegedy. 'Intriguing properties of neural networks'. In: *arXiv preprint arXiv:1312.6199* (2013).

[45] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. rndi, P. Laskov, G. Giacinto and F. Roli. 'Evasion attacks against machine learning at test time'. In: *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III 13*. Springer. 2013, pp. 387–402.

[46] I. J. Goodfellow, J. Shlens and C. Szegedy. 'Explaining and harnessing adversarial examples'. In: *arXiv preprint arXiv:1412.6572* (2014).

[47] S.-M. Moosavi-Dezfooli, A. Fawzi and P. Frossard. 'Deepfool: a simple and accurate method to fool deep neural networks'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2574–2582.

[48] N. Carlini and D. Wagner. 'Towards Evaluating the Robustness of Neural Networks'. In: *arXiv preprint arXiv:1608.04644* (Aug. 2016).

[49] A. Madry. 'Towards deep learning models resistant to adversarial attacks'. In: *arXiv preprint arXiv:1706.06083* (2017).

[50] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi and P. Frossard. 'Universal adversarial perturbations'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1765–1773.

[51] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi and C.-J. Hsieh. 'Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models'. In: *Proceedings of the 10th ACM workshop on artificial intelligence and security*. 2017, pp. 15–26.

[52] A. Ilyas, L. Engstrom and A. Madry. 'Prior Convictions: Black-box Adversarial Attacks with Bandits and Priors'. In: *International Conference on Learning Representations*. 2019. 2019.

[53] A. Ilyas, L. Engstrom, A. Athalye and J. Lin. 'Black-box adversarial attacks with limited queries and information'. In: *International conference on machine learning*. PMLR. 2018, pp. 2137–2146.

[54] S. Moon, G. An and H. O. Song. 'Parsimonious black-box adversarial attacks via efficient combinatorial optimization'. In: *International conference on machine learning*. PMLR. 2019, pp. 4636–4645.

[55] N. Narodytska and S. P. Kasiviswanathan. 'Simple Black-Box Adversarial Attacks on Deep Neural Networks.' In: *CVPR Workshops*. Vol. 2. 2. 2017.

[56] W. Brendel, J. Rauber and M. Bethge. 'Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models'. In: *International Conference on Learning Representations*. 2018.

[57] J. Chen, M. I. Jordan and M. J. Wainwright. 'Hopskipjumpattack: A query-efficient decision-based attack'. In: *2020 ieee symposium on security and privacy (sp)*. IEEE. 2020, pp. 1277–1294.

[58] M. Cheng, T. Le, P.-Y. Chen, H. Zhang, J. Yi and C.-J. Hsieh. 'Query-Efficient Hard-label Black-box Attack: An Optimization-based Approach'. In: *International Conference on Learning Representations*. 2019.

[59] M. Cheng, S. Singh, P. H. Chen, P.-Y. Chen, S. Liu and C.-J. Hsieh. 'Sign-OPT: A Query-Efficient Hard-label Adversarial Attack'. In: *International Conference on Learning Representations*. 2020.

[60] N. Papernot, P. McDaniel and I. Goodfellow. 'Transferability in machine learning: from phenomena to black-box attacks using adversarial samples'. In: *arXiv preprint arXiv:1605.07277* (2016).

[61] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik and A. Swami. 'Practical black-box attacks against machine learning'. In: *Proceedings of the 2017 ACM on Asia conference on computer and communications security*. 2017, pp. 506–519.

[62] Y. Liu, X. Chen, C. Liu and D. Song. 'Delving into transferable adversarial examples and black-box attacks'. In: *arXiv preprint arXiv:1611.02770* (2016).

[63] F. Tramèr, N. Papernot, I. Goodfellow, D. Boneh and P. McDaniel. 'The space of transferable adversarial examples'. In: *arXiv preprint arXiv:1704.03453* (2017).

[64] X. Wang and K. He. 'Enhancing the transferability of adversarial attacks through variance tuning'. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 1924–1933.

[65] T. Gu, K. Liu, B. Dolan-Gavitt and S. Garg. 'Badnets: Evaluating backdooring attacks on deep neural networks'. In: *IEEE Access* 7 (2019), pp. 47230–47244.

[66] M. Barni, K. Kallas and B. Tondi. 'A new backdoor attack in cnns by training set corruption without label poisoning'. In: *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2019, pp. 101–105.

[67] Y. Liu, X. Ma, J. Bailey and F. Lu. 'Reflection backdoor: A natural backdoor attack on deep neural networks'. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*. Springer. 2020, pp. 182–199.

[68] 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. 11966–11976.

[69] Y. Li, Y. Li, B. Wu, L. Li, R. He and S. Lyu. 'Invisible backdoor attack with sample-specific triggers'. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 16463–16472.

[70] T. A. Nguyen and A. T. Tran. 'WaNet - Imperceptible Warping-based Backdoor Attack'. In: *International Conference on Learning Representations*. 2021.

[71] W. Jiang, H. Li, G. Xu and T. Zhang. '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.

[72] H. A. A. K. Hammoud and B. Ghanem. 'Check your other door! establishing backdoor attacks in the frequency domain'. In: (2021).

[73] T. Wang, Y. Yao, F. Xu, S. An, H. Tong and T. Wang. 'An invisible black-box backdoor attack through frequency domain'. In: *European Conference on Computer Vision*. Springer. 2022, pp. 396–413.

[74] Y. Zeng, W. Park, Z. M. Mao and R. Jia. 'Rethinking the backdoor attacks' triggers: A frequency perspective'. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 16473–16481.

[75] R. Hou, T. Huang, H. Yan, L. Ke and W. Tang. 'A stealthy and robust backdoor attack via frequency domain transform'. In: *World Wide Web* (2023), pp. 1–17.

[76] Y. Gao, H. Chen, P. Sun, J. Li, A. Zhang, Z. Wang and W. Liu. 'A dual stealthy backdoor: From both spatial and frequency perspectives'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 38. 3. 2024, pp. 1851–1859.

[77] K. Doan, Y. Lao and P. Li. 'Backdoor attack with imperceptible input and latent modification'. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 18944–18957.

[78] N. Zhong, Z. Qian and X. Zhang. 'Imperceptible Backdoor Attack: From Input Space to Feature Representation'. In: *International Joint Conference on Artificial Intelligence*. 2022.

[79] Z. Zhao, X. Chen, Y. Xuan, Y. Dong, D. Wang and K. Liang. '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, pp. 15213–15222.

[80] S. Cheng, Y. Liu, S. Ma and X. Zhang. 'Deep feature space trojan attack of neural networks by controlled detoxification'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 2. 2021, pp. 1148–1156.

[81] A. Turner, D. Tsipras and A. Madry. ‘Label-consistent backdoor attacks’. In: *arXiv preprint arXiv:1912.02771* (2019).

[82] Y. Zeng, M. Pan, H. A. Just, L. Lyu, M. Qiu and R. Jia. ‘Narcissus: A Practical Clean-Label Backdoor Attack with Limited Information’. In: *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*. CCS ’23. New York, NY, USA: Association for Computing Machinery, 2023, pp. 771–785. ISBN: 9798400700507.

[83] A. Saha, A. Subramanya and H. Pirsiavash. ‘Hidden trigger backdoor attacks’. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. 07. 2020, pp. 11957–11965.

[84] H. Souri, L. Fowl, R. Chellappa, M. Goldblum and T. Goldstein. ‘Sleeper agent: Scalable hidden trigger backdoors for neural networks trained from scratch’. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 19165–19178.

[85] A. Vassilev, A. Oprea, A. Fordyce and H. Andersen. ‘Adversarial machine learning: A taxonomy and terminology of attacks and mitigations’. In: (2024).

[86] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra. ‘Grad-cam: Visual explanations from deep networks via gradient-based localization’. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 618–626.

[87] G. Baruch, M. Baruch and Y. Goldberg. ‘A little is enough: Circumventing defenses for distributed learning’. In: *Advances in Neural Information Processing Systems* 32 (2019).

[88] A. N. Bhagoji, S. Chakraborty, P. Mittal and S. Calo. ‘Analyzing federated learning through an adversarial lens’. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 634–643.

[89] C. Xie, K. Huang, P.-Y. Chen and B. Li. ‘Dba: Distributed backdoor attacks against federated learning’. In: *International Conference on Learning Representations*. 2019.

[90] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin and V. Shmatikov. ‘How to backdoor federated learning’. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 2938–2948.

[91] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee and D. Papailiopoulos. ‘Attack of the tails: Yes, you really can backdoor federated learning’. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 16070–16084.

[92] H. Li, Q. Ye, H. Hu, J. Li, L. Wang, C. Fang and J. Shi. ‘3dfed: Adaptive and extensible framework for covert backdoor attack in federated learning’. In: *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2023, pp. 1893–1907.

[93] H. Zhang, J. Jia, J. Chen, L. Lin and D. Wu. ‘A3fl: Adversarially adaptive backdoor attacks to federated learning’. In: *Advances in Neural Information Processing Systems* 36 (2024).

[94] Z. Zhang, A. Panda, L. Song, Y. Yang, M. Mahoney, P. Mittal, R. Kannan and J. Gonzalez. 'Neurotoxin: Durable Backdoors in Federated Learning'. In: *Proceedings of the 39th International Conference on Machine Learning*. 2022, pp. 26429–26446.

[95] Y. Dai and S. Li. 'Chameleon: Adapting to peer images for planting durable backdoors in federated learning'. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 6712–6725.

[96] S. Shen, S. Tople and P. Saxena. 'Auror: Defending against Poisoning Attacks in Collaborative Deep Learning Systems'. In: *Proceedings of the 32nd Annual Conference on Computer Security Applications*. ACSAC '16. Los Angeles, California, USA: Association for Computing Machinery, 2016, pp. 508–519. ISBN: 9781450347716.

[97] S. Li, M. Xue, B. Z. H. Zhao, H. Zhu and X. Zhang. 'Invisible backdoor attacks on deep neural networks via steganography and regularization'. In: *IEEE Transactions on Dependable and Secure Computing* 18.5 (2020), pp. 2088–2105.

[98] K. D. Doan, Y. Lao and P. Li. 'Marksman Backdoor: Backdoor Attacks with Arbitrary Target Class'. In: *arXiv preprint arXiv:2210.09194* (2022).

[99] Y. Shi, M. Du, X. Wu, Z. Guan, J. Sun and N. Liu. 'Black-box Backdoor Defense via Zero-shot Image Purification'. In: *Advances in Neural Information Processing Systems*. 2023, pp. 57336–57366.

[100] K. Zhang, G. Tao, Q. Xu, S. Cheng, S. An, Y. Liu, S. Feng, G. Shen, P.-Y. Chen, S. Ma and X. Zhang. 'FLIP: A Provable Defense Framework for Backdoor Mitigation in Federated Learning'. In: *The Eleventh International Conference on Learning Representations*. 2023.

[101] Y. Wang, H. Su, B. Zhang and X. Hu. 'Interpret neural networks by identifying critical data routing paths'. In: *proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 8906–8914.

[102] J. Carnerero-Cano, L. Muñoz-González, P. Spencer and E. C. Lupu. 'Hyperparameter Learning under Data Poisoning: Analysis of the Influence of Regularization via Multiobjective Bilevel Optimization'. In: *arXiv preprint arXiv:2306.01613* (2023).

[103] Z. Sun, P. Kairouz, A. T. Suresh and H. B. McMahan. 'Can you really backdoor federated learning?' In: *arXiv preprint arXiv:1911.07963* (2019).

[104] A. Panda, S. Mahloujifar, A. N. Bhagoji, S. Chakraborty and P. Mittal. 'Sparsefed: Mitigating model poisoning attacks in federated learning with sparsification'. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2022, pp. 7587–7624.

[105] T. D. Nguyen, P. Rieger, H. Chen, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen, A. Mirhoseini, S. Zeitouni, F. Koushanfar, A.-R. Sadeghi and T. Schneider. 'FLAME: Taming Backdoors in Federated Learning'. In: *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 1415–1432. ISBN: 978-1-939133-31-1.

[106] P. Rieger, T. D. Nguyen, M. Miettinen and A.-R. Sadeghi. 'DeepSight: Mitigating backdoor attacks in federated learning through deep model inspection'. In: *NDSS*. 2022.

[107] D. Yin, Y. Chen, R. Kannan and P. Bartlett. 'Byzantine-robust distributed learning: Towards optimal statistical rates'. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 5650–5659.

[108] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli and A. Anandkumar. 'signSGD: Compressed Optimisation for Non-Convex Problems'. In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by J. Dy and A. Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, 2018, pp. 560–569.

[109] A. Kurakin, I. J. Goodfellow and S. Bengio. 'Adversarial Machine Learning at Scale'. In: *5th International Conference on Learning Representations* (2016).

[110] S. M. Moosavi-Dezfooli, A. Fawzi and P. Frossard. 'DeepFool: a simple and accurate method to fool deep neural networks'. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2015), pp. 2574–2582.

[111] J. Su, D. V. Vargas and S. Kouichi. 'One pixel attack for fooling deep neural networks'. In: *IEEE Transactions on Evolutionary Computation* 23.5 (2017), pp. 828–841.

[112] M. Andriushchenko, F. Croce, N. Flammarion and M. Hein. 'Square Attack: a query-efficient black-box adversarial attack via random search'. In: *Lecture Notes in Computer Science* 12368 LNCS (2019), pp. 484–501.

[113] I. J. Goodfellow, J. Shlens and C. Szegedy. 'Explaining and Harnessing Adversarial Examples'. In: (2014).

[114] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow and R. Fergus. 'Intriguing properties of neural networks'. In: (2013).

[115] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui and M. I. Jordan. 'Theoretically Principled Trade-off between Robustness and Accuracy'. In: (2019).

[116] A. Madry, A. Makelov, L. Schmidt, D. Tsipras and A. Vladu. 'Towards Deep Learning Models Resistant to Adversarial Attacks'. In: (2017).

[117] O. Poursaeed, T. Jiang, H. Yang, S. Belongie and S.-N. Lim. *Robustness and Generalization via Generative Adversarial Training*. Tech. rep.

[118] C. Laidlaw, S. Singla and S. Feizi. 'Perceptual Adversarial Robustness: Defense Against Unseen Threat Models'. In: (2020).

[119] J. Tack, S. Yu, J. Jeong, M. Kim, S. J. Hwang and J. Shin. 'Consistency Regularization for Adversarial Robustness'. In: *AAAI Conference on Artificial Intelligence* 36 (2021), pp. 8414–8422.

[120] E. Wong, L. Rice and J. Z. Kolter. 'Fast is better than free: Revisiting adversarial training'. In: (2020).

[121] V. Srinivasan, A. Marban, K.-R. Müller, W. Samek and S. Nakajima. 'Robustifying Models Against Adversarial Attacks by Langevin Dynamics'. In: (2018).

[122] C. Shi, C. Holtz and G. Mishne. ‘Online Adversarial Purification based on Self-Supervision’. In: (2021).

[123] Y. Yang, G. Zhang, D. Katabi and Z. Xu. ‘ME-Net: Towards Effective Adversarial Robustness with Matrix Estimation’. In: (2019).

[124] F. Tramèr, N. Carlini and W. Brendel. *On Adaptive Attacks to Adversarial Example Defenses*. Tech. rep.

[125] S. Chen, Z. Huang, Q. Tao, Y. Wu, C. Xie and X. Huang. *Adversarial Attack on Attackers: Post-Process to Mitigate Black-Box Score-Based Query Attacks*. Tech. rep.

[126] J. Wang, Z. Lyu, D. Lin, B. Dai and H. Fu. ‘Guided Diffusion Model for Adversarial Purification’. In: (2022).

[127] J. Ho, A. Jain and P. Abbeel. ‘Denoising Diffusion Probabilistic Models’. In: (2020).

[128] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon and B. Poole. ‘Score-Based Generative Modeling through Stochastic Differential Equations’. In: (2020).

[129] A. Vahdat, K. Kreis and J. Kautz. *Score-based Generative Modeling in Latent Space*. Tech. rep.

[130] P. Dhariwal and A. Nichol. *Diffusion Models Beat GANs on Image Synthesis*. Tech. rep.

[131] Z. He, A. S. Rakin and D. Fan. *Parametric Noise Injection: Trainable Randomness to Improve Deep Neural Network Robustness against Adversarial Attack*. Tech. rep.

[132] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat and A. Anandkumar. ‘Diffusion Models for Adversarial Purification’. In: (2022).

[133] G. Lin, C. Li, J. Zhang, T. Tanaka and Q. Zhao. ‘Adversarial Training on Purification (AToP): Advancing Both Robustness and Generalization’. In: (2024).

[134] D. Stutz, M. Hein and B. Schiele. ‘Confidence-Calibrated Adversarial Training: Generalizing to Unseen Attacks’. In: *International Conference on Machine Learning* (2019), pp. 9092–9103.

2

BACKDOOR ATTACK AGAINST DEEP NEURAL NETWORKS

Convolutional Neural Networks (CNNs) that have excelled in diverse computer vision tasks are vulnerable to backdoor attacks, enabling attacker-controlled predictions via specific triggers. *Restricted to spatial domains, recent research exploits perceptual traits by embedding triggers in the frequency domain, yielding pixel-level indistinguishable perturbations. In black-box settings, restricted access to model and training process necessitates advanced trigger designs. Current frequency-based attacks manipulate magnitude spectra, introducing discrepancies between clean and poisoned data, though vulnerable to common image processing operations like compression and filtering.*

*In this paper, we propose a robust low-frequency backdoor attack (**LFBA**) in black-box setup that minimally perturbs spectrum components and maintains the perceptual similarity in spatial space simultaneously. Our methodology capitalizes on the insight that optimal triggers can be located in low-frequency regions to maximize attack effectiveness, robustness against image transformation operations, and stealthiness in* dual *space. To effectively explore the discrete frequency space, we utilize simulated annealing (SA), a form of evolutionary algorithm, to optimize the properties of trigger including the frequency bands to be manipulated and the perturbation of each band under restricted attack scenario. Extensive experiments on both CNNs and Vision Transformers (ViT) confirm the effectiveness and robustness of LFBA against image processing operations and state-of-the-art backdoor defenses. Furthermore, LFBA exhibits inherent stealthiness in both spatial and frequency spaces, making it resistant to human and frequency inspection.*

This chapter is based on the paper “Low-Frequency Black-Box Backdoor Attack via Evolutionary Algorithm" by Qiao, Y., Liu, D., Wang, R., Liang, K. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2025.

2.1. INTRODUCTION

CNNs are vulnerable to backdoor attacks [1–6] that can mislead the model to make attack-chosen predictions with *triggers* in the use phase while behaving normally on clean images, causing severe consequences in high-stakes applications such as autonomous driving [7] and biometric authentication [8].

Prior backdoor attacks possess the capability to inject imperceptible triggers into spatial domain [11–17]. Inserting triggers in spatial space can harm the semantics of infected image pixels (see fig. 2.1). Recent works have concluded that backdoor attacks can inject trigger patterns into frequency space [9, 10, 18, 19]. For example, FTrojan [9] manipulates mid- and high-frequency spectrum of input images with a pre-defined perturbation within fixed frequency band. However, manually crafting frequency components, especially in high-frequency regions, could harm the robustness of trigger and thus trigger effectiveness can be eliminated by image processing operations such as lowpass filters. Moreover, both spatial and frequency triggers introduce distinguishable artifacts when transformed to frequency space (see figs. 2.1 and 2.6).

Inspired by [20], an ideal and practical backdoor attack should achieve four objectives, namely, *functionality preservation*, *effectiveness*, *dual-space stealthiness*, and *robustness*. Functionality preservation ensures high test accuracy on clean data. Effectiveness is demonstrated by the ability to misclassify poisoned data to the target label with a high probability. Dual-space stealthiness implies that poisoned images exhibit visual and frequency similarity to clean ones. Robustness is demonstrated by its effectiveness against image transformations and resistance to backdoor defenses. Although successfully achieving the goals at pixel level, [20] does not consider the stealthiness in the frequency perspective. This work explores a new perspective of backdoors in frequency domain.

Typically, low-frequency components of an image contain semantic information, while high-frequency components capture finer details and noise. According to prior works such as [21, 22], inserting triggers in low-frequency region offers concrete influences: (1) low-frequency components have a perceptual capacity that allows trigger insertion without perceptual degradation; (2) low-frequency components exhibit greater resilience in lossy compression operations such as JPEG, whereas high-frequency components are more pronounced to data loss; and (3) trigger inserted in low-frequency region is harder to be removed by low-pass filtering compared to the high-frequency region.

Building on the above insights, we develop LFBA, a poisoning-based low-frequency backdoor attack with a robust and imperceptible trigger in dual space. The key insight of LFBA is to find the optimal trigger that can achieve dual-space stealthiness, attack effectiveness and robustness when the model and defense strategy are unknown. This design marks the first exploration into the robustness from a frequency perspective, which locates mininal perturbations in low-frequency region against image transformations while naturally guaranteeing perceptual similarity in the pixel domain. Finding such a frequency trigger in a black-box scenario is not trivial. Due to the absence of the victim model and training process, one may handcraft the frequency trigger as FTrojan [9] and FIBA [10], but it

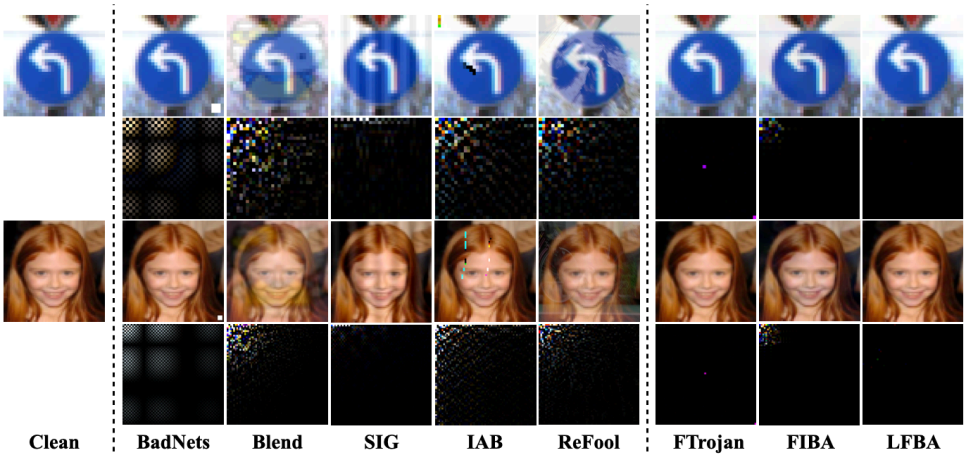


Figure 2.1: Comparison of poisoned images with their corresponding frequency disparities (amplified by 5×) to clean images of existing attacks. **Left**: clean images; **mid**: poisoned images of spatial-based attacks including BadNets [3], Blend [2], SIG [1], IAB [6] and ReFool [5]; **right**: poisoned images of frequency-based attacks including FTrojan [9], FIBA [10] and our LFBA. Although state-of-the-art frequency triggers achieve superior invisibility in pixel space than spatial triggers, they introduce anomaly frequency artifacts.

could lead to improper frequency properties of the trigger. For example, a large perturbation can disrupt invisibility of poisoned images, while a small perturbation could hinder the model's ability to learn the feature of trigger, resulting in low attack effectiveness. Furthermore, selecting an improper frequency band for trigger insertion can compromise the attack robustness against image processing operations (see Table 2.8).

To address the challenges under black-box setting, we leverage simulated annealing (SA), an effective gradient-free optimization algorithm, to search for the optimal trigger in the discrete spectrum space. Specifically, we convert a clean image between spatial and frequency domain via discrete cosine transform (DCT) and its inverse (IDCT). Then, we inject our frequency trigger into the spectrum of the image and iteratively optimize two properties of the trigger including perturbations and frequency bands via SA, in order to maximize attack effectiveness as the primary goal with a penalty term concerning dual-space stealthiness. To efficiently estimate the trigger effectiveness, we use the objective loss from a semi-trained surrogate model. Finally, we use our frequency trigger function to produce attacker's poisoned dataset with the optimal trigger. Since the imperceptible perturbations are posed in low-frequency region, LFBA preserves the invisibility in spatial domain and is robust to any eradication in frequency domain. The **main contributions** of this work are as follows:

- We explore a new perspective of frequency trigger design that provides the

dual-space stealthiness and robustness, and develop a practical low-frequency backdoor attack method.

• We formulate a constrained optimization problem to find optimal frequency triggers. Then we utilize SA to effectively minimize the attack objectives of the problem.

• Extensive experiments in computer vision tasks empirically demonstrate that the proposed attack provides state-of-the-art effectiveness and robustness against existing spatial and frequency defenses as well as image transformation operations.

2.2. RELATED WORK

Backdoor Attacks. [3] introduces the first backdoor attack against deep learning models. It employs a patch-based pattern as trigger, injecting it into a small fraction of clean data, which causes the victim model to misclassify those poisoned images to the target label. After that, various attacks, aiming at improving stealthiness and robustness through the design of triggers and training process, have been proposed in the literature.

(1) *Spatial-based attacks*: To enhance the invisibility of triggers for bypassing human inspection, some works [1, 5, 11–13, 20] focus on imperceptible backdoor attacks in spatial domain. For example, [1] uses sinusoidal signals as triggers which results in slightly varying backgrounds; [5] utilizes natural reflection as trigger into the victim model, while [12] leverages DNN-based image steganography technique to hide an attacker-specified string into clean images as sample-specific triggers. Later, several works [14–17] reveal the importance of stealthiness in latent feature space. [14] learns a trigger generator to constrain the similarity of hidden features between poisoned and clean data via Wasserstein regularization. To improve the stealthiness of triggers, [16] adaptively learns the generator by constraining the latent layers, which makes triggers more invisible in both input and latent feature space. While spatial attacks offer inherent stealthiness, they often overlook robustness against common image processing operations utilized during data preprocessing. Consequently, their effectiveness is significantly compromised by such operations. Moreover, most attacks require a strong attack assumption that the adversary possesses full control over the training process and has knowledge of the victim model. More importantly, many spatial backdoor attacks exhibit severe high-frequency artifacts that can be easily detected in the frequency domain (see figs. 2.1 and 2.6).

(2) *Frequency-based attacks*: Recent works [9, 10, 18, 19, 23, 24] explore another attack surface, namely, frequency domain, naturally guaranteeing invisibility due to frequency properties. FTrojan [9] handcrafts two single frequency bands with fixed perturbations as trigger, and FIBA [10] injects low-frequency information of a trigger image by linearly combining spectral amplitude of poisoned and clean images. These works introduce distinguishable frequency artifacts and can be detected via frequency inspection. DUBA [24] embeds high-frequency information of trigger image into clean image to achieve trigger invisibility in dual domains. However, it focuses on stealthiness and do not consider robustness against image transformation operations. Different from the above works, we propose a black-box

frequency backdoor attack that firstly achieves imperceptibility in dual spaces and robust against image processing defenses. *We briefly compare the SOTA backdoor attacks in Table 2.1 based on various attack attributes.*

Backdoor Defenses. Defensive [26–32] and detective [23, 33–36] mechanisms are commonly used for backdoor defenses. Defensive methods focus on mitigating the effectiveness of potential backdoor attacks. For example, fine-pruning [26] prunes the dormant neurons in the last convolution layer based on clean inputs' activation values. Neural Cleanse [27] reconstructs potential triggers for each target label via reverse engineering and renders the backdoor ineffective by retrain patches strategy. Neural Attention Distillation [31] uses a "teacher" model to guide the finetuning of the backdoored "student" network to erase backdoor triggers. Representative detective methods include STRIP [33] which perturbs or superimposes clean inputs to identify the potential backdoors during inference time, spectral signature [35] using latent feature representations to detect outliers and [23] leveraging supervised learning to differentiate between clean and poisoned data in frequency space. Besides, image processing-based methods [9, 32, 37] have been studied, which remove backdoors by image processing transformations. In this work, we showcase that the proposed attack can evade representative defenses including frequency inspection, image processing operations and mainstream backdoor defenses.

Recently, several state-of-the-art backdoor defenses have been proposed. For example, ASD [38] introduces a training-time defense that separates training data into clean and poisoned subsets. Neural Polarizer [39] purifies poisoned models by incorporating a learnable neural polarizer as an intermediate layer. ZIP [40] mitigates backdoor attacks through zero-shot image purification.

Threat Model. We consider a rather realistic black-box scenario as in prior works [9, 20, 41] where the adversary, i.e. a malicious data provider, can only inject a limited number of poisoned samples into clean training set for public use. The attacker should not have control over the training process or have knowledge of the victim model. This is a more practical and challenging attack scenario than white-box attacks [11, 14, 16, 17, 42]. Such a threat model can be seen in many real-world scenarios like the outsourcing of data collection to third-parties.

2.3. PROPOSED METHOD

2.3.1. PRELIMINARIES

Backdoor attacks.. We consider backdoor attacks on image classification. Let $f_\theta : \mathscr{I} \rightarrow \mathbb{R}^K$ be an image classifier parameterized with θ that maps an input image $\mathscr{I} \subseteq [0,1]^{H\times W\times C}$ to an output class, where K is the number of classes, H, W and C are the height, width and channels of an input image. The parameters θ of the classifier are learned using a training dataset $D_c = \{(x_i, y_i)|x_i \in \mathscr{I}, y_i \in \mathbb{R}^K\}_{i=1}^{N}$.

In a standard backdoor attack, the attacker crafts a subset of D_c with ratio ρ to produce the poisoned dataset $D_p = \{(x'_i, y'_i)|x'_i \in \mathscr{I}, y'_i \in \mathbb{R}^K\}_{i=1}^{N\times\rho}$ by the trigger function $\mathscr{T}$ and target label function η. Given a clean image x from the clean subset and its true class y, the commonly used trigger function $\mathscr{T}$ and target label funtion η in the spatial space are defined with a hyper-parameter $m \in [0,1]$ and a trigger pattern t as

Table 2.1: Critical attack attributes among LFBA and other attacks in spatial (S) and frequency (F) domains.

Attributes→ Attacks ↓	Attack Domain	Attack Scenario	Stealthiness		Attack Robustness	Optimization
			S	F		
IAB [6]	S	White-box	✗	✗	✗	w/
ISSBA [12]	S	White-box	✗	✗	✗	w/
LIRA [11]	S	White-box	✗	✗	✗	w/
DFST [17]	S	White-box	✗	✗	✗	w/
WB [14]	S	White-box	✗	✗	✗	w/
IBA [15]	S	White-box	✗	✗	✗	w/
BadNets [3]	S	Black-box	✗	✗	✗	w/o
SIG [1]	S	Black-box	✓	✗	✗	w/o
ReFool [5]	S	Black-box	✓	✗	✗	w/o
WaNet [13]	S	Black-box	✓	✗	✗	w/o
Narcissus [25]	S	Black-box	✗	✗	✓	w/
FTrojan [9]	F	Black-box	✓	✗	✗	w/o
FIBA [10]	F	Black-box	✓	✗	✗	w/o
DUBA [24]	S+F	Black-box	✓	✓	✗	w/o
LFBA (Ours)	S+F	Black-box	✓	✓	✓	w/

follows:

$$x' = \mathcal{T}(x, m, t) = x \cdot (1 - m) + t \cdot m, \quad y' = \eta(y) = y_t, \tag{2.1}$$

where y_t is the target class. Under empirical risk minimization, a typical attack aims to inject backdoors into the classifier f by learning θ with both clean dataset D_c and poisoned dataset D_p so that the classifier misclassifies the poisoned data into the target class while behaving normally on clean data. The optimization problem is defined as follows:

$$\min_{\theta} \sum_{(x,y)\in D_c} \mathcal{L}(f_\theta(x), y) + \sum_{(x',y')\in D_p} \mathcal{L}(f_\theta(x'), y'), \tag{2.2}$$

where $\mathcal{L}$ represents the cross-entropy loss.

DCT and IDCT functions. DCT is a widely used transformation that represents a finite sequence of image pixels as a sum of cosine functions oscillating at various frequencies. In the spectrum, most of the semantic information of images tends to be concentrated in a few low-frequency components on the top-left region, where the $(0,0)$ element (top-left) is the zero-frequency component. Given an image $x(h, w, c)$, its frequency form $x^f(h^f, w^f, c)$ is calculated by the DCT function $\mathcal{D}(\cdot)$ as

follows:

$$x^f(h^f, w^f, c) = \mathscr{D}(x(h, w, c)) \tag{2.3}$$

$$= V(h^f)V(w^f)\sum_{h=0}^{H-1}\sum_{w=0}^{W-1}\sum_{c=0}^{C-1} x(h, w, c)\cos\left[\frac{(2h+1)h^f\pi}{2H}\right]\cos\left[\frac{(2w+1)w^f\pi}{2W}\right] \tag{2.4}$$

for $\forall h, h^f = 0, 1, ..., H-1$ and $\forall w, w^f = 0, 1, ..., W-1$, where H, W, C represent the height, width and number of channels of the given image. For simplicity, we assume $H = W$, therefore $V(0) = \sqrt{\frac{1}{4H}}$ and $V(k) = \sqrt{\frac{1}{2H}}$ for $k > 0$. Accordingly, $\mathscr{D}^{-1}(\cdot)$ denotes the IDCT as follows:

$$x(h, w, c) = \mathscr{D}^{-1}(x^f(h^f, w^f, c)) \tag{2.5}$$

$$= \sum_{h^f=0}^{H-1}\sum_{w^f=0}^{W-1}\sum_{c=0}^{C-1} V(h)V(w)x^f(h^f, w^f, c)\cos\left[\frac{(2h^f+1)h\pi}{2H}\right]\cos\left[\frac{(2w^f+1)w\pi}{2W}\right] \tag{2.6}$$

2.3.2. FREQUENCY BACKDOOR ATTACK

Frequency Trigger Function. We redesign the trigger function in the frequency space to better search the frequency trigger that can achieve dual-space stealthiness. Given a clean sample $(x, y) \in D_c$, we first transform x to the frequency domain via DCT function $\mathscr{D}(\cdot)$ and obtain the frequency spectrum x^f. Next, we apply our frequency trigger function $\mathscr{T}^f$ to insert the trigger t into x^f. Subsequently, we revert it to the spatial domain via IDCT function $\mathscr{D}^{-1}(\cdot)$ and manipulate the ground truth label y to the desired target y_t to obtain the poisoned sample (x', y'). The process can be formulated as:

$$x' = \mathscr{T}^f(x, t) \triangleq \mathscr{D}^{-1}(\mathscr{D}(x) + t), \quad y' = \eta(y) = y_t, \tag{2.7}$$

where the trigger $t = (\delta, \nu)$ comprises a set of perturbations $\delta = \{\delta_i | i = 1, 2, \ldots, n\}$ in terms of frequency components and its corresponding frequency bands $\nu = \{\nu_i | i = 1, 2, \ldots, n\}$ that indicates the position in frequency spectrum to pose the perturbation δ on, n represents the number of frequency bands to manipulate. Finally, we apply $\mathscr{T}^f$ to a subset of D_c with ratio ρ to produce D_p.

Problem Formulation. In contrast to white-box attacks, which manipulate the training process of the victim model, our primary goal is to search the optimal trigger (δ^*, ν^*) that can achieve high attack effectiveness. To mitigate the absence of victim model and the extensive time consumption of training from scratch, we follow [20, 25] to fine-tune a semi-trained surrogate model f_θ^s with the poisoned dataset D_p for a few epochs. We use the training loss of D_p from f_θ^s to approximate the trigger effectiveness with an acceptable deviation in practice. Therefore, the main task of our attack to minimize is defined as follows:

$$O(\delta, \nu) = \sum_{(x,y)\in D_p} \mathscr{L}(f_\theta^s(\mathscr{T}^f(x, (\delta, \nu)), y_t). \tag{2.8}$$

One may argue that large crafted perturbations in specific frequency bands could also achieve high attack effectiveness and practical natural stealthiness such as

[9, 10]. Triggers without careful consideration can bring distinguishable frequency artifacts (see figs. 2.1 and 2.6). We hereby define a penalty term for dual-space stealthiness to ensure imperceptible frequency perturbations as follows:

$$P(\delta, \nu) = \|\delta\|_p, \tag{2.9}$$

where $p = 2$ denotes l_2-norm distance of perturbations on frequency components[1]. Taking into account both the primary goal and the dual-space stealthiness penalty, our attack goal aims to minimize the overall objective function under the constraints w.r.t. magnitude of perturbations and region of manipulated bands. The optimization problem of LFBA is formulated as follows:

$$\min_{\delta,\nu} \quad O(\delta, \nu) + P(\delta, \nu), \tag{2.10}$$

$$\text{s.t.} \quad |\delta_i| \leq \epsilon, \tag{2.11}$$

$$\nu_i \in \mathscr{F}_{low}, \tag{2.12}$$

where $i \in \{0, 1, \cdots, n-1\}$, n is the number of manipulated bands, ϵ constrains the maximal value of each perturbation δ_i, and $\mathscr{F}_{low}$ is the low-frequency region for searching LFBA trigger.

2.3.3. FREQUENCY TRIGGER OPTIMIZATION

It is challenging to apply common gradient-based methods, such as stochastic gradient descent (SGD), to optimize discrete variables (e.g., frequency bands) in our context. Therefore, we search the optimal trigger with simulated annealing (SA) [43], a probabilistic optimization technique, to effectively optimize δ and ν in discrete spectral space. Algorithm 1 describes the workflow of searching our optimal trigger $t^* = (\delta^*, \nu^*)$ with SA in low-frequency region $\mathscr{F}_{low}$.

Particularly, the optimization process starts with randomly initializing a trigger $t_{opt} = (\delta_{opt}, \nu_{opt})$, which satisfies two hard constraints in eqs. (2.11) and (2.12). Then, we inject the subset of D_c with t_{opt} to produce D_p. After that, we train the surrogate model f_θ^s on D_p for a few epochs E and approximate the effectiveness of trigger based on objective value Obj_{opt} computed using eq. (2.10). The annealing process involves heating a material to a high temperature and then gradually cooling it to remove defects and optimize its internal structure. To reflect this process, we decrease the temperature T from initial temperature T_0 to terminal temperature T_f by a decay factor α to control the trigger optimization process. As shown in Algorithm 2, under each T, SA iteratively generates an offspring $t = (\delta, \nu)$ restricted by eqs. (2.11) and (2.12) to improve the trigger effectiveness and stealthiness. Then we follow a similar process of initialization to calculate the attack objective value Obj for t in eq. (2.10). Once t is better than t_{opt} in terms of the objective value, i.e., $Obj < Obj_{opt}$, the current outperformed trigger t and Obj will survive as the new t_{opt} and Obj_{opt} and then enter the next round, otherwise the above process will be repeated. Upon termination (T drops to T_f), the last t_{opt} is the desired optimal trigger t^* that is used by eq. (2.7) to produce the attacker's poisoned dataset.

[1] The l_2-norm distance of trigger perturbations between frequency and spatial domains remains consistent, i.e., $\|\delta\|_2 \equiv \|\mathscr{D}^{-1}(\delta)\|_2$. We hereby only consider the frequency domain.

Algorithm 1: Optimal Frequency Trigger Search via SA

Require: Poisoned dataset D_p, Initial temperature T_0, Terminal temperature T_f, Optimization iterations per temperature *iter*, Annealing factor α, Number of epochs E, Surrogate model f_θ^s, Maximum frequency perturbation ϵ, Low frequency region $\mathcal{F}_{low}$
Ensure: The optimal frequency trigger $t^* = (\delta^*, \nu^*)$
$Step1: Initialization$
$\delta_{opt} \leftarrow$ Rand$(-\epsilon, \epsilon)$, $\nu_{opt} \leftarrow$ Rand_Loc$(\mathcal{F}_{low})$
Poison D_p with $t_{opt} = (\delta_{opt}, \nu_{opt})$ using $\mathcal{T}^f$ in eq. (2.7)
$f_{\theta'}^s \leftarrow$ Train f_θ^s on D_p within E
$Obj_{opt} \leftarrow O(\delta_{opt}, \nu_{opt}) + P(\delta_{opt}, \nu_{opt})$ by eq. (2.10) with D_p on $f_{\theta'}^s$
$Step2: Trigger\ optimization$
$T = T_0$
while T $\geq T_f$ **do**
  for $i = 1, 2, \ldots, iter$ **do**
    $(\delta, \nu) \leftarrow TriggerUpdate((\delta_{opt}, \nu_{opt}), T)$ // see Algorithm 2
    Poison D_p with $t = (\delta, \nu)$ using $\mathcal{T}^f$ in eq. (2.7)
    $f_{\theta'}^s \leftarrow$ Train f_θ^s on D_p within E
    $Obj \leftarrow O(\delta, \nu) + P(\delta, \nu)$ by eq. (2.10) with D_p on $f_{\theta'}^s$
    if $Obj < Obj_{opt}$ **then**
      $\delta_{opt} \leftarrow \delta, \nu_{opt} \leftarrow \nu$, $Obj_{opt} \leftarrow Obj$
    end
  end
  $T = T$ - $\alpha \times T$
end
$\delta^* \leftarrow \delta_{opt}, \nu^* \leftarrow \nu_{opt}$
return $t^* = (\delta^*, \nu^*)$

2.4. EXPERIMENTS

2.4.1. EXPERIMENTAL SETUP

Datasets and Models. Without loss of generality, we evaluate LFBA on five benchmark datasets including MNIST [44], CIFAR-10 [45], Tiny-ImageNet (T-IMNET) [46], GTSRB [47] and CelebA [48]. For CelebA, we follow [13, 41] to select the top three most balanced attributes including Heavy Makeup, Mouth Slightly Open, and Smiling. Then we concatenate them to create an eight-label classification task. We test LFBA on both small and large-scale datasets with a wide range of image sizes, including both grayscale and RGB images, to verify attack performance and also remain consistency across different types of image datasets. Following [13, 34, 35, 42, 49], we consider various architectures for the victim image classifier. Specifically, we employ a CNN model [13, 42] for MNIST, PreAct-ResNet18 [50] for CIFAR-10

Algorithm 2: Trigger Update Method

Require: Current trigger (δ_{opt}, v_{opt}), Mutation probability $prob_{mut}$, Initial temperature T_0, Current temperature T, Terminal temperature T_f, Number of manipulated frequency bands n
Ensure: The evolved trigger
(δ, v) $\delta \leftarrow \delta_{opt}, v \leftarrow v_{opt}$
if $Rand(0,1) < prob_{mut}$ **then**
  $i \leftarrow \text{Rand}(0, n)$
  $\Delta v \leftarrow \text{Rand\_Loc}(\mathscr{F}_{low})$
  $v_i \leftarrow v_i + (T - T_f)/(T_0 - T_f) * \Delta v$
  $j \leftarrow \text{Rand}(0, n)$
  $\Delta\delta \leftarrow \text{Rand}(-\epsilon, \epsilon)$
  $\delta_j \leftarrow \delta_j + (T - T_f)/(T_0 - T_f) * \Delta\delta$
end
return (δ, v)

and GTSRB, and ResNet18 [50] for T-IMNET and CelebA. To demonstrate the generalization of LFBA, we evaluate its effectiveness on state-of-the-art architectures for vision tasks, called Vision Transformers (ViT) [51]. The detailed ViT architectures used in our experiments are provided in Table 2.2. For surrogate models, we use heterogeneous VGG architectures [52] to simulate black-box setup. The details of tasks, datasets and models are described in Table 2.3.

Table 2.2: Overview of parameter setups of ViT architectures.

Settings ↓	ViT (CIFAR-10)	ViT (CelebA)
hidden_dim	128	512
num_layers	6	6
num_heads	4	8
image_size	32	64
patch_size	4	4
mlp_dim	256	512
drop_out	0.1	0.1
num_parameters	0.81M	9.63M

Table 2.3: The summary of tasks, and their corresponding models.

Task	Dataset	# of Training/Test Images	# of Labels	Image Size	Victim Model	Surrogate Model
Handwritten Digit Recognition	MNIST	60,000/10,000	10	28×28×1	3 Conv + 2 Dense	VGG11
Object Classification	CIFAR-10	50,000/10,000	10	32×32×3	PreAct-ResNet18 / ViT	VGG16
Traffic Sign Recognition	GTSRB	39,209/12,630	43	32×32×3	PreAct-ResNet18	VGG16
Object Classification	Tiny-ImageNet	100,000/10,000	200	64×64×3	ResNet18	VGG19
Face Attribute Recognition	CelebA	162,770/19,962	8	64×64×3	ResNet18 / ViT	VGG19

Evaluation Metrics. We evaluate attack effectiveness based on attack success rate (ASR), i.e. the ratio of poisoned samples successfully misclassified to the target label, and test accuracy (ACC) on clean data for functionality-preserving requirement. For human inspection, we use PSNR [53], SSIM [54] and LPIPS [55] to evaluate spatial invisibility between clean and poisoned data. LPIPS utilizes deep features of CNNs to identify perceptual similarity, while SSIM and PSNR are calculated based on the statistical pixel-wise similarity.

Implementation. The implementation of LFBA is based on PyTorch [56] and executed on a workstation with 16-core AMD Ryzen 9 7950X CPU, NVIDIA GeForce RTX 4090 and 64G RAM. For the default setting, we train CNN models by SGD optimizer with learning rate 0.01 and decayed by a factor of 0.1 after every 50 epochs. For ViT architectures, we utilize AdamW optimizer [57] with learning rate 0.01. We set batch size to 64 and total number of epochs to 200. We set ϵ to 0.1 for MNIST, 0.5 for CIFAR-10, GTSRB, and 1.5 for T-IMNET and CelebA. Following the approach outlined in [58], we select approximately 18.3% of the frequency spectrum in the top-left region for $\mathscr{F}_{low}$ to search for our trigger. We choose $n = 3$ as the number of manipulated frequency bands. For simplicity, we set the poison ratio to only 5% and target label to 7 for all the datasets[2]. For trigger optimization, we set the parameters in SA as follows: $iter = 5$, $T_0 = 1$, $T_f = 0.01$, $\alpha = 0.99$ and $prob_{mut} = 0.8$. Unless explicitly stated otherwise, we adopt this default setting for LFBA in the experimental sections.

2.4.2. ATTACK PERFORMANCE

We compare both spatial and frequency attacks including BadNets [3], SIG [1], ReFool [5], WaNet [13], FTrojan [9] and FIBA [10] as baselines to evaluate the effectiveness. Since other backdoor attacks [11, 14–17] require full control over training process and knowledge of the victim classifiers, we do not consider the above methods as practical baselines.

Attack effectiveness. *We first demonstrate that LFBA achieves high ASR (≥ 99%) across 5 datasets and 3 models with slight accuracy degradation (<0.55% in average) (see Table 2.4).* The results confirm that our attack outperforms other black-box attacks in most tasks. That is so because we approximate the trigger effectiveness during the optimization process of SA (see eq. (2.10)) whereas others do not take into account attack effectiveness when designing their triggers. Additionally, LFBA also achieves the highest ASR compared to other spatial and frequency backdoors on ViT. For example, LFBA achieves a 99.38% ASR, which is 1.08% higher than BadNets on CelebA. *Our experimental findings raise urgent concerns for the physical realm: adversaries can compromise any model by injecting publicly available images with a robust and dual-space stealthy trigger, even without access to the victim model.*

Computational cost of trigger optimization. To illustrate the practical applicability of the selected optimization method in a real-world scenario, we evaluate the computational overhead of trigger optimization using SA. Table 2.5 showcases the

[2] Our attack is label-independent, i.e., the attacker can easily transfer LFBA attack to any other desired labels by searching the corresponding optimal triggers.

Table 2.4: Attack performance via ACC (%) and ASR (%) for several attacks.

Attack	MNIST		GTSRB		CIFAR-10		Tiny-ImageNet		CelebA	
	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR
CLEAN	99.33	-	98.60	-	93.25	-	55.24	-	79.13	-
BADNETS	99.25	**99.99**	98.05	97.16	92.05	98.24	54.09	97.82	76.54	99.35
SIG	99.31	99.95	97.90	99.87	92.14	**99.98**	54.59	99.49	77.90	99.85
REFOOL	98.71	99.28	97.94	98.51	91.09	97.03	54.37	97.32	78.53	98.09
WANET	98.59	99.09	98.19	99.83	92.31	99.93	54.85	99.16	77.99	99.33
FTROJAN	99.27	99.94	96.63	99.25	92.53	99.82	53.41	99.38	78.63	99.20
FIBA	99.27	99.74	96.73	98.88	91.13	97.87	54.11	98.14	77.90	99.16
LFBA	**99.31**	99.72	**98.42**	**99.97**	**92.91**	99.88	**54.64**	**99.90**	**78.79**	**99.91**

searching time to generate the optimal frequency trigger for each dataset. We can see that SA achieves a reasonable optimization time, averaging around tens of seconds. Therefore, SA is a suitable choice for our optimization method in LFBA.

Table 2.5: The computational cost of trigger optimization via SA across different datasets.

Dataset	MNIST	GTSRB	CIFAR-10	T-IMNET	CelebA
Time	5 s	61 s	39 s	35 s	192 s

Table 2.6: Attack performance via ACC (%) and ASR (%) for several attacks on ViT.

Attack	CIFAR-10		CelebA	
	ACC	ASR	ACC	ASR
CLEAN	85.41	-	73.55	-
BADNETS	83.15	99.86	72.77	98.30
SIG	83.19	99.89	72.68	98.99
REFOOL	82.97	98.46	72.45	97.24
WANET	83.28	99.72	72.67	98.75
FTROJAN	**83.52**	99.91	72.51	99.02
FIBA	83.30	99.85	72.40	98.66
LFBA	83.41	**99.99**	**72.87**	**99.38**

Natural stealthiness. A dual-space stealthiness penalty is added to the process of searching the optimal LFBA trigger so as to ensure natural stealthiness of poisoned images. We compare the state-of-the-art invisible attacks in spatial and frequency domains. For each dataset, we randomly select 500 sample images from test dataset to evaluate trigger stealthiness. A higher PSNR/SSIM or a smaller LPIPS value indicates a better stealthiness of an given poisoned sample. LFBA achieves more natural stealthiness than current frequency backdoor attacks (see Table 4.4)

due to its less number of frequency bands and minimal perturbations. Such minor alterations of LFBA in frequency space can naturally provide invisibility to the potential defender who lacks knowledge of the correspondent clean image. Taking Table 4.4 and Figure 2.1 into consideration, we conclude that *the proposed LFBA attack outperforms both spatial and frequency domain-based attacks in terms of natural stealthiness.*

Table 2.7: Natural stealthiness (PSNR ↑, SSIM ↑, LPIPS ↓).

Attacks	CIFAR-10			Tiny-ImageNet			CelebA		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
CLEAN	Inf	1.0000	0.0000	Inf	1.0000	0.0000	Inf	1.0000	0.0000
BADNETS	36.67	0.9763	0.0012	36.35	0.9913	0.0006	32.50	0.9951	0.0005
SIG	25.26	0.8533	0.0289	25.36	0.8504	0.0631	25.38	0.7949	0.0359
REFOOL	18.37	0.6542	0.0697	20.42	0.8564	0.4574	23.72	0.8359	0.2134
WANET	19.30	0.8854	0.0090	29.59	0.9359	0.0360	30.42	0.9175	0.0530
FTROJAN	41.16	0.9946	0.0006	42.28	0.9931	0.0003	42.25	0.9904	0.0002
FIBA	29.69	0.9858	0.0024	29.39	0.9755	0.0080	29.25	0.9592	0.0057
LFBA	**44.31**	**0.9971**	**0.0001**	**43.54**	**0.9942**	**0.0002**	**46.27**	**0.9953**	**0.0001**

2.4.3. ATTACK AGAINST DEFENSIVE MEASURES

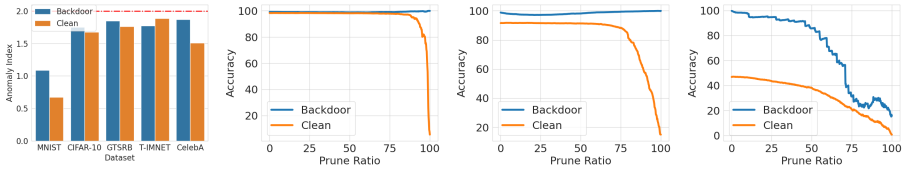
We evaluate attack robustness of LFBA against the mainstream defenses including Neural Cleanse [27], STRIP [33], Fine-pruning [26] and network inspection. We also show imperceptible frequency artifacts of LFBA against frequency artifacts inspection [23]. Moreover, we evaluate our attack under preprocessing-based defenses as in [9, 20] to comprehensively illustrate the practical robustness.

Neural Cleanse (NC). The key intuition of NC is that a backdoor trigger can cause any input misclassified to target label. It reverses engineering possible triggers and detects backdoors in the victim model using anomaly index. An anomaly index exceeding 2 signifies that the model has been compromised. LFBA remains below the threshold and successfully evades the defense across all datasets (see Figure 2.2 (a)). We recall that NC focuses on small and fixed patches but LFBA designs trigger in frequency space, wherein it inserts an imperceptible frequency perturbation only causing a minimal change in the entire pixel domain. Consequently, trigger spans the entire pixel space, providing considerable natural similarity.

STRIP. It assumes that the predictions given by a backdoored model on poisoned samples consistently tend to be target label and are not easily changed. It detects poisoned samples by analyzing the classification entropy after superimposing some random clean samples on the test samples. We can observe that LFBA achieves almost the same entropy probability distributions as clean samples (see Figure 2.3), allowing it to circumvent the defense. The overlap area of distributions refers to the difficulty of poisoned sample detection. For example, the distributions of clean and poisoned samples on CIFAR-10 are almost indistinguishable, indicating that it is hard for STRIP to detect our poisoned samples. This is so because superimposing random

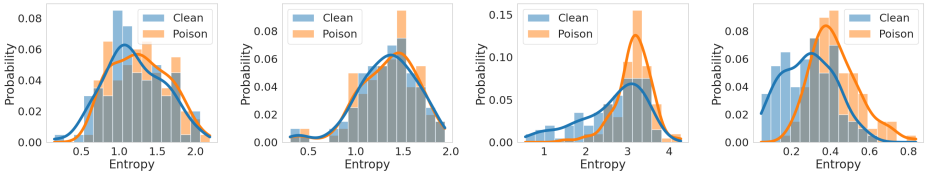
images in spatial space destroy low-frequency components (containing LFBA trigger pattern) of poisoned images. Therefore, the predictions of superimposed images will also undergo significant changes, resembling the clean case.

Fine-pruning. It mitigates backdoor effectiveness by pruning dominant neurons with very low activations via a small clean dataset. We test LFBA against Fine-Pruning and demonstrate ACC and ASR with respect to the ratio of pruned neuron on GTSRB, CIFAR-10 and T-IMNET (see Figure 2.2 (b)-(d)). Across all datasets, we see that the ASR is always higher than ACC without any degradation, making backdoor mitigation impossible. This suggests that Fine-pruning is ineffective against LFBA.



(a) NC (b) GTSRB (c) CIFAR-10 (d) T-IMNET

Figure 2.2: (a): The results of LFBA under NC on different datasets; (b)-(d): The attack effectiveness of LFBA against Fine-pruning.



(a) GTSRB (b) CIFAR-10 (c) T-IMNET (d) CelebA

Figure 2.3: The entropy distributions of LFBA against STRIP under 4 datasets.

Network Inspection. We further investigate the impact of LFBA on the attention of the classifier by Grad-CAM [59]. Grad-CAM finds the critical regions of input images that mostly activate models prediction. In Figure 2.4, we showcase visual heatmaps of clean and poisoned images by LFBA. We observe that LFBA does not introduce anomaly attention areas of networks when compared to clean cases across all datasets. This is because our trigger is inserted in low-frequency components, which contain the semantics of images, making network attention unaltered.

Image Preprocessing-based Defenses. We select image preprocessing methods in [9, 20], including Gaussian filter, Wiener filter, BM3D [60] and JPEG compression [61], which directly denoise or compress input images. We further apply these operations to poisoned test images of CIFAR-10 with various hyperparameters before inference. The results are shown in Table 2.8. They demonstrate that all transformations are effective to remove trigger effectiveness in FTrojan, which handcrafts mid- and high-frequency components with anomaly perturbations. However, LFBA can

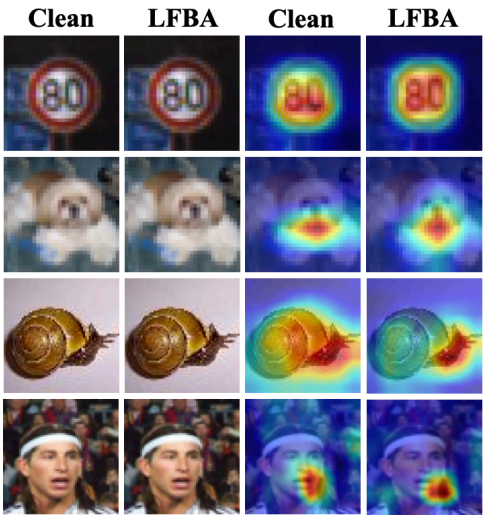
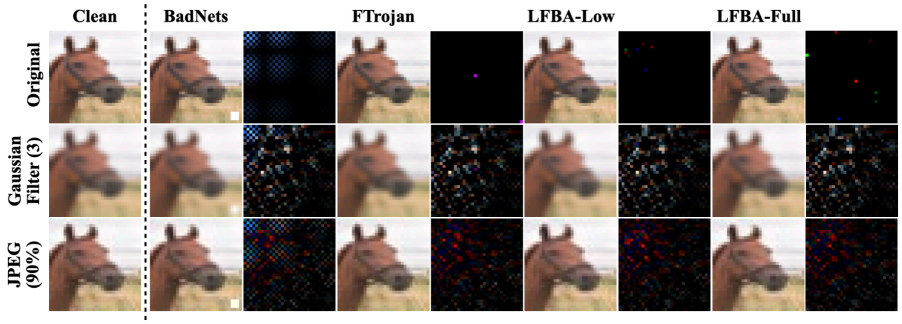


Figure 2.4: Visualization of network attention on GTSRB, CIFAR-10, T-IMNET and CelebA. Compared to the feature saliency maps of clean images, LFBA does not introduce any unusual regions.

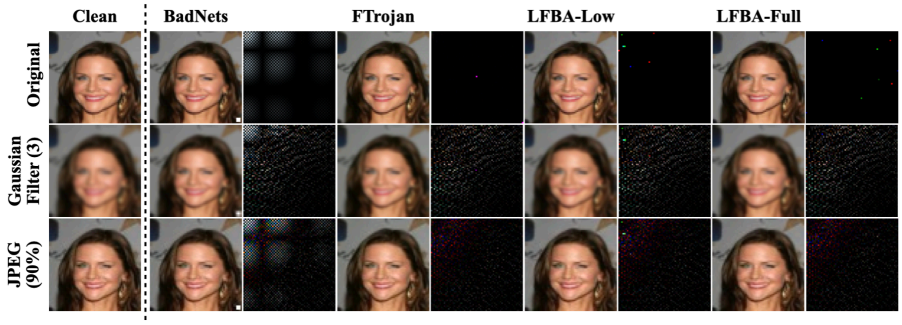
circumvent these defenses since denoising transformations and lossy compression do not typically operate on the low-frequency components [22].

Explanations of Robustness through Frequency Perspective. We showcase poisoned images and their frequency disparities (compared to clean images) under the image transformations in Figure 2.5. We can see that the frequency disparities of BadNets remain similar to the original ones after JPEG compression while the Gaussian filter destroys the BadNets patterns on both datasets. This proves the fact, as shown in Table 2.8, that BadNets is effective against JPEG compression but fails to survive after Gaussian filtering. For FTrojan and LFBA-Full, we cannot see any frequency patterns after these transformations. However, the frequency disparities of LFBA-Low can be clearly seen even after such operations, indicating our low-frequency attack is robust against preprocessing-based defenses. We note that low-frequency components exhibit greater resilience to image transformations than mid- and high-frequency components.

Frequency Artifacts Inspection. We consider the same frequency artifacts inspection method as in [23]. In Figure 2.6, we compare the frequency spectrum between clean and poisoned images and calculate l_2-norm distance between them. We can see that current spatial backdoors introduce more anomaly artifacts than frequency backdoors. It is worth noting that two spikes lie in central and bottom-right regions in FTrojan's spectrum and anomaly perturbations in SIG's. However, LFBA spectrums closely resemble those of clean images on both datasets (exhibiting the smallest l_2-norm distances and similarly smooth spectral distributions as clean samples). According to [23, 62, 63], our poisoned samples exhibit the same frequency properties as natural images due to dual-space stealthiness. Therefore, frequency



(a) CIFAR-10



(b) CelebA

Figure 2.5: Comparison of poisoned images with their corresponding frequency disparities (amplified by 5×) to clean images of existing attacks under different image preprocessing-based defenses. Each frequency disparities spectrum is calculated based on the original clean image's spectrum. These image transformations can effectively remove the trigger pattern through frequency domain, while the disparities spectrums of our LFBA-Low attack still contain original backdoor patterns.

inspection is ineffective to detect anomaly artifacts of LFBA.

In conclusion, a wide range of experimental results empirically demonstrate that *LFBA can elude or significantly degrade the performance of the state-of-the-art defenses in dual space* even when both the model and defense strategy are unknown. Besides, our frequency trigger is *resilient to image preprocessing-based defenses, which provides more robustness than existing attacks.* The results also indicate the pivotal role of LFBA in bolstering the security of machine learning systems.

2.4.4. ABLATION STUDY

We here analyze several hyperparameters that are critical for the LFBA performance.
Frequency stealthiness constraint ϵ and number of manipulated band n. ϵ restrains the maximum perturbation of each frequency band while n controls the number of

Table 2.8: Attack robustness of various triggers against preprocessing-based defenses. To illustrate the robustness of our low-frequency trigger, we introduce a full-spectrum variant of LFBA for comparison, named LFBA-Full, which searches the trigger across the entire spectrum with same attack settings.

Attacks →	BadNets		FTrojan		LFBA-Low		LFBA-Full	
Methods ↓	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR
Original	92.02	98.78	92.53	99.82	92.91	99.88	91.21	99.73
Gaussian Filter ($w = 3$)	66.17	15.11	67.80	6.47	72.23	98.52	71.46	7.81
Gaussian Filter ($w = 5$)	39.81	6.88	45.03	3.25	53.54	97.27	49.07	3.59
Wiener Filter ($w = 3$)	69.53	96.02	69.11	10.54	71.92	98.22	71.61	6.93
Wiener Filter ($w = 5$)	52.18	90.81	49.20	5.28	52.03	95.65	50.90	3.66
BM3D ($\sigma = 0.5$)	87.39	98.44	87.34	15.84	88.31	99.09	87.08	13.58
BM3D ($\sigma = 1.0$)	86.03	94.07	86.40	19.33	86.70	98.04	86.64	18.69
JPEG (quality = 90%)	88.98	97.85	89.22	9.36	82.72	89.75	89.18	9.54
JPEG (quality = 50%)	78.84	92.59	79.66	8.58	76.19	75.93	76.31	8.54
Average ASR		73.97		9.83		**94.06**		9.04

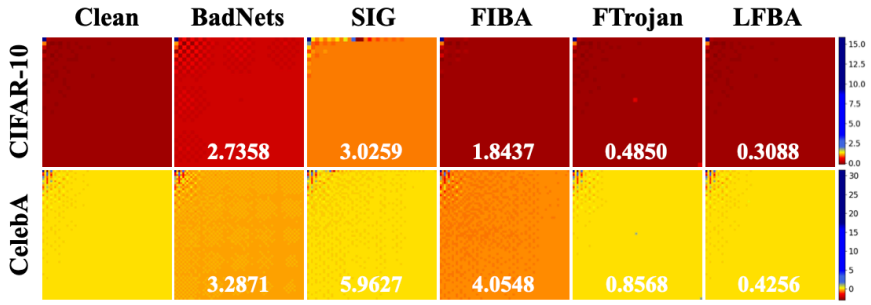


Figure 2.6: Visualization of averaged DCT spectrum results under various spatial and frequency black-box attacks on 10,000 randomly selected samples from two datasets of different image sizes.

manipulated frequency bands. We visualize the impact of ϵ and n on the poisoned images (see Figure 2.7). If ϵ and n are set too large, the poisoned image may be easily recognized (i.e., lacking stealthiness) upon human inspection in the pixel domain, which could also introduce distinguishable frequency disparities. On the other hand, setting ϵ and n too small results in the trigger having a low proportion of features in dual spaces. In this sense, the classifier will encounter difficulty in catching and learning these trigger features, yielding a drop of attack effectiveness. Figure 2.8 illustrates the influences of ϵ and n on the attack effectiveness among the tasks. The ASRs decline significantly and eventually fall below 20% as we continuously decrease ϵ to 0.01, in which evidences can be seen in GTSRB under various n, while there is a drastic drop occurs from $\epsilon = 0.5$ to $\epsilon = 0.1$ in CIFAR-10. We notice that increasing n enhances the effectiveness of LFBA. For instance, the

ASR increases from 77% to 92.8% under $\epsilon = 0.1$ when n increases from 1 to 4 in GTSRB. We also observe that a large ϵ allows single injection in frequency band to achieve a high ASR. For instance, selecting $\epsilon = 1$ and $n = 1$ to perform our attack can achieve nearly 100% ASR. However, such an attack setup could compromise frequency stealthiness. Thus, it's crucial to consider a balance between dual-space stealthiness and attack effectiveness before conducting a LFBA attack.

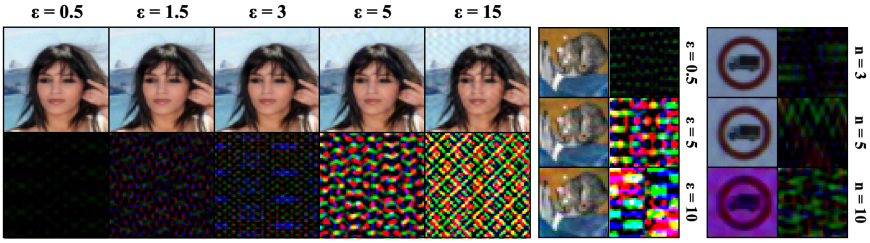
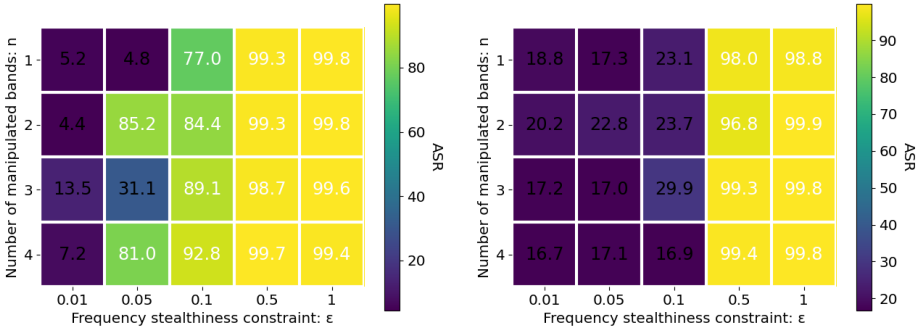


Figure 2.7: Visualization of LFBA poisoned images and triggers under different ϵ and n. The pixel value of triggers is amplified by 30×.

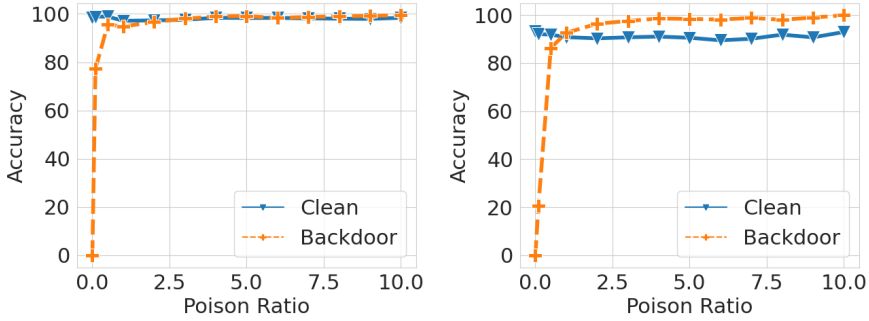


(a) GTSRB (b) CIFAR-10

Figure 2.8: The impact of ϵ and n.

Poison ratio ρ. ρ is the fraction of poisoned samples in the training dataset of the adversary. We test the attack effectiveness under different ρ varying from 0.1% to 10%. Although we increase ρ from a wide range, LFBA does not harm the ASR of the victim models. As stated in Figure 2.9, this fraction setting cannot degrade the ACC and meanwhile, we would like to examine the lower bound of the fraction that LFBAs effectiveness can withstand. Even when ρ is 0.1%, LFBA can still provide a high ASR, around 80% for GTSRB. We also find that sensitivities to poison ratio can vary among tasks. In CIFAR-10, LFBA achieves above 86% ASR under $\rho = 0.5\%$ while it drops rapidly, around 20%, when ρ reduces to 0.1%.

Transferability studies. We test LFBA's transferability on CIFAR-10 dataset across a wide range of typical model architectures including ViT, GoogLeNet [64], ResNet18



(a) GTSRB (b) CIFAR-10

Figure 2.9: The impact of attack effectiveness under a wide range of poison ratios (%).

and VGG16 from small to large size (see Table 2.9 for the number of model parameters). We use each surrogate-victim model pair to search trigger and train the poisoned model.

Table 2.9: Overview of total parameters of surrogate and victim models.

Model	Number of parameters
ViT	0.81 M
GoogleNet	6.80 M
ResNet18	11.69 M
VGG16	138.37 M

In Table 2.10, we first verify that the attack effectiveness is not harmed by the surrogate-victim model mismatch and attains high ASRs (> 99%) for all model pairs. We also observe that having the same surrogate and victim models does not always result in the best ASR. Additionally, a larger size of surrogate architecture does not necessarily maximize the attack effectiveness. For example, using GoogLeNet as the surrogate model which is smaller than ResNet18 can provide the best ASR of 99.45%. In conclusion, the attacker could deliver a successful attack without detailed information about the victim model.

2.5. CONCLUSION AND DISCUSSION

We propose a robust black-box backdoor attack by inserting imperceptible perturbations in the low-frequency domain. Compared to current works, LFBA for the first offers superior stealthiness in dual domains and robustness against image transformations. We leverage SA to effectively optimize the trigger in the discrete spectrum space to achieve four attack objectives. The empirical experiments demonstrate that LFBA can achieve a practical attack robustness to evade SOTA

Table 2.10: Transferrability of LFBA across different surrogate-victim model architecture pairs via ACC (%) and ASR (%) on CIFAR-10. LFBA provides practical transferability between surrogate and victim models when estimating the effectiveness of trigger.

Victim →	ViT		GoogLeNet		ResNet18		VGG16	
Surrogate ↓	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR
ViT	82.75	99.68	93.61	**99.43**	92.79	99.29	91.08	99.33
GoogLeNet	82.11	99.58	93.31	99.01	93.27	**99.45**	91.79	99.19
ResNet18	82.63	**99.91**	**93.69**	99.23	93.23	99.37	91.70	**99.41**
VGG16	**83.12**	99.43	93.16	99.08	**93.66**	99.19	**92.04**	99.37

defenses in both spatial and frequency domains as well as image transformations.
Discussion. In this work, we concentrate on various computer vision tasks, which have been the focus of numerous existing works [9, 11, 13, 41, 42]. In the future, we intend to expand the scope of this work to other vision tasks, e.g., objection detections and semantic segmentations, and other SOTA model architectures, e.g., diffusion models. The trigger search process is executed in a hybrid GPU-CPU environment during trigger evaluation and optimization phases. It deserves further efforts to design a GPU-accelerated SA to minimize data transmission across hardware, thus improving the efficiency of our proposed LFBA. Note that black-box attacks such as LFBA fail to achieve the same level of robustness against state-of-the-art backdoor defenses as white-box methods due to the lack of control over the training process of the victim model. To further enhance the robustness against those defenses, one would combine advanced training mechanisms proposed in white-box attacks with our frequency trigger to develop a more stealthy and robust backdoor attack that can bypass countermeasures.

2.6. ETHICAL CONSIDERATION

This work exposes the vulnerability of deep learning models to practical, stealthy and robust backdoors and can inspire follow-up studies that enhance the security of machine learning systems. In this sense, this work has a positive impact on the future research of AI safety. In the following, we discuss intellectual property, intended usage, potential misuse, risk control and human subjects.
Intellectual property. All comparative attacks and defenses, models, datasets and implementation libraries are open-source. We believe that the datasets are well-desensitized. We strictly comply with all applicable licenses for academic use.
Intended Usage. We expose the vulnerability of current centralized deep learning models to dual-space stealthy and robust backdoor triggers. We encourage researchers to use our findings to assess the security of their models and hope that this work will inspire the development of robustness against such backdoor attacks.
Potential Misuse. This work could be exploited to produce harmful poisoned datasets for real-world applications, which potentially leads to more covertly

malicious models. To maintain the security of deep learning models, we will propose an adaptive defense in the future.

Risk Control. To further mitigate potential risks, we will release the code used in this work. By doing so, we believe that transparency will reduce the risks related to our work, encourage responsible use and foster further advancement of secure techniques for centralized deep learning systems.

Human Subjects. We do not involve any human subjects in this work. Instead, we rely solely on mathematical and model-based metrics to simulate human visual inspection, thereby eliminating the need for human participation.

REFERENCES

[1] M. Barni, K. Kallas and B. Tondi. 'A new backdoor attack in cnns by training set corruption without label poisoning'. In: *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2019, pp. 101–105.

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

[3] T. Gu, K. Liu, B. Dolan-Gavitt and S. Garg. 'Badnets: Evaluating backdooring attacks on deep neural networks'. In: *IEEE Access* 7 (2019), pp. 47230–47244.

[4] Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang and X. Zhang. 'Trojaning attack on neural networks'. In: *25th Annual Network And Distributed System Security Symposium (NDSS 2018)*. Internet Soc. 2018.

[5] Y. Liu, X. Ma, J. Bailey and F. Lu. 'Reflection backdoor: A natural backdoor attack on deep neural networks'. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*. Springer. 2020, pp. 182–199.

[6] T. A. Nguyen and A. Tran. 'Input-aware dynamic backdoor attack'. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 3454–3464.

[7] S. Grigorescu, B. Trasnea, T. Cocias and G. Macesanu. 'A survey of deep learning techniques for autonomous driving'. In: *Journal of Field Robotics* 37.3 (2020), pp. 362–386.

[8] H. Heidari and A. Chalechale. 'Biometric authentication using a deep learning approach based on different level fusion of finger knuckle print and fingernail'. In: *Expert Syst. Appl.* (2022).

[9] T. Wang, Y. Yao, F. Xu, S. An, H. Tong and T. Wang. 'An invisible black-box backdoor attack through frequency domain'. In: *European Conference on Computer Vision*. Springer. 2022, pp. 396–413.

[10] Y. Feng, B. Ma, J. Zhang, S. Zhao, Y. Xia and D. Tao. 'Fiba: Frequency-injection based backdoor attack in medical image analysis'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 20876–20885.

[11] 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. 11966–11976.

[12] Y. Li, Y. Li, B. Wu, L. Li, R. He and S. Lyu. 'Invisible backdoor attack with sample-specific triggers'. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 16463–16472.

[13] T. A. Nguyen and A. T. Tran. 'WaNet - Imperceptible Warping-based Backdoor Attack'. In: *International Conference on Learning Representations*. 2021.

[14] K. Doan, Y. Lao and P. Li. 'Backdoor attack with imperceptible input and latent modification'. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 18944–18957.

[15] N. Zhong, Z. Qian and X. Zhang. 'Imperceptible Backdoor Attack: From Input Space to Feature Representation'. In: *International Joint Conference on Artificial Intelligence*. 2022.

[16] Z. Zhao, X. Chen, Y. Xuan, Y. Dong, D. Wang and K. Liang. '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, pp. 15213–15222.

[17] S. Cheng, Y. Liu, S. Ma and X. Zhang. 'Deep feature space trojan attack of neural networks by controlled detoxification'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 2. 2021, pp. 1148–1156.

[18] H. A. A. K. Hammoud and B. Ghanem. 'Check your other door! establishing backdoor attacks in the frequency domain'. In: (2021).

[19] R. Hou, T. Huang, H. Yan, L. Ke and W. Tang. 'A stealthy and robust backdoor attack via frequency domain transform'. In: *World Wide Web* (2023), pp. 1–17.

[20] W. Jiang, H. Li, G. Xu and T. Zhang. '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.

[21] I. Cox, J. Kilian, F. Leighton and T. Shamoon. 'Secure spread spectrum watermarking for multimedia'. In: *IEEE Transactions on Image Processing* 6.12 (1997), pp. 1673–1687.

[22] C. Guo, J. S. Frank and K. Q. Weinberger. 'Low frequency adversarial perturbation'. In: *arXiv preprint arXiv:1809.08758* (2018).

[23] Y. Zeng, W. Park, Z. M. Mao and R. Jia. 'Rethinking the backdoor attacks' triggers: A frequency perspective'. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 16473–16481.

[24] Y. Gao, H. Chen, P. Sun, J. Li, A. Zhang, Z. Wang and W. Liu. 'A dual stealthy backdoor: From both spatial and frequency perspectives'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 38. 3. 2024, pp. 1851–1859.

[25] Y. Zeng, M. Pan, H. A. Just, L. Lyu, M. Qiu and R. Jia. 'Narcissus: A Practical Clean-Label Backdoor Attack with Limited Information'. In: *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*. CCS '23. New York, NY, USA: Association for Computing Machinery, 2023, pp. 771–785. ISBN: 9798400700507.

[26] 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.

[27] 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.

[28] H. Chen, C. Fu, J. Zhao and F. Koushanfar. 'DeepInspect: A Black-box Trojan Detection and Mitigation Framework for Deep Neural Networks.' In: *IJCAI*. Vol. 2. 5. 2019, p. 8.

[29] X. Qiao, Y. Yang and H. Li. 'Defending neural backdoors via generative distribution modeling'. In: *Advances in neural information processing systems* 32 (2019).

[30] P. Zhao, P.-Y. Chen, P. Das, K. N. Ramamurthy and X. Lin. 'Bridging mode connectivity in loss landscapes and adversarial robustness'. In: *arXiv preprint arXiv:2005.00060* (2020).

[31] Y. Li, X. Lyu, N. Koren, L. Lyu, B. Li and X. Ma. 'Neural attention distillation: Erasing backdoor triggers from deep neural networks'. In: *arXiv preprint arXiv:2101.05930* (2021).

[32] Y. Li, T. Zhai, B. Wu, Y. Jiang, Z. Li and S. Xia. 'Rethinking the Trigger of Backdoor Attack'. In: *CoRR* (2020). arXiv: 2004.04692.

[33] 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.

[34] B. Chen, W. Carvalho, N. Baracaldo, H. Ludwig, B. Edwards, T. Lee, I. Molloy and B. Srivastava. 'Detecting backdoor attacks on deep neural networks by activation clustering'. In: *arXiv preprint arXiv:1811.03728* (2018).

[35] B. Tran, J. Li and A. Madry. 'Spectral signatures in backdoor attacks'. In: *Advances in neural information processing systems* 31 (2018).

[36] S. Kolouri, A. Saha, H. Pirsiavash and H. Hoffmann. 'Universal litmus patterns: Revealing backdoor attacks in cnns'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 301–310.

[37] H. Qiu, Y. Zeng, S. Guo, T. Zhang, M. Qiu and B. Thuraisingham. 'Deepsweep: An evaluation framework for mitigating DNN backdoor attacks using data augmentation'. In: *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*. 2021, pp. 363–377.

[38] K. Gao, Y. Bai, J. Gu, Y. Yang and S.-T. Xia. 'Backdoor defense via adaptively splitting poisoned dataset'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 4005–4014.

[39] M. Zhu, S. Wei, H. Zha and B. Wu. 'Neural Polarizer: A Lightweight and Effective Backdoor Defense via Purifying Poisoned Features'. In: *Advances in Neural Information Processing Systems*. Vol. 36. 2023, pp. 1132–1153.

[40] Y. Shi, M. Du, X. Wu, Z. Guan, J. Sun and N. Liu. 'Black-box Backdoor Defense via Zero-shot Image Purification'. In: *Advances in Neural Information Processing Systems*. 2023, pp. 57336–57366.

[41] 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.

[42] K. D. Doan, Y. Lao and P. Li. 'Marksman Backdoor: Backdoor Attacks with Arbitrary Target Class'. In: *arXiv preprint arXiv:2210.09194* (2022).

[43] P. J. Van Laarhoven, E. H. Aarts, P. J. van Laarhoven and E. H. Aarts. *Simulated annealing*. Springer, 1987.

[44] Y. LeCun and C. Cortes. 'The mnist database of handwritten digits'. In: 2005.

[45] A. Krizhevsky, G. Hinton *et al.* 'Learning multiple layers of features from tiny images'. In: (2009).

[46] Y. Le and X. Yang. 'Tiny imagenet visual recognition challenge'. In: *CS 231N* 7.7 (2015), p. 3.

[47] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing and C. Igel. 'Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark'. In: *The 2013 international joint conference on neural networks (IJCNN)*. Ieee. 2013, pp. 1–8.

[48] Z. Liu, P. Luo, X. Wang and X. Tang. 'Deep Learning Face Attributes in the Wild'. In: *Proceedings of International Conference on Computer Vision (ICCV)*. 2015.

[49] R. Shokri *et al.* 'Bypassing backdoor detection algorithms in deep learning'. In: *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE. 2020, pp. 175–183.

[50] 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.

[51] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.* 'An image is worth 16x16 words: Transformers for image recognition at scale'. In: *arXiv preprint arXiv:2010.11929* (2020).

[52] K. Simonyan and A. Zisserman. 'Very Deep Convolutional Networks for Large-Scale Image Recognition'. In: *International Conference on Learning Representations*. 2015.

[53] Q. Huynh-Thu and M. Ghanbari. 'Scope of validity of PSNR in image/video quality assessment'. In: *Electronics letters* 13 (2008), pp. 800–801.

[54] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli. 'Image quality assessment: from error visibility to structural similarity'. In: *IEEE transactions on image processing* 13.4 (2004), pp. 600–612.

[55] R. Zhang, P. Isola, A. A. Efros, E. Shechtman and O. Wang. 'The unreasonable effectiveness of deep features as a perceptual metric'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 586–595.

[56] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala. 'PyTorch: An Imperative Style, High-Performance Deep Learning Library'. In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035.

[57] I. Loshchilov and F. Hutter. 'Decoupled weight decay regularization'. In: *arXiv preprint arXiv:1711.05101* (2017).

[58] Y. Sharma, G. W. Ding and M. Brubaker. 'On the effectiveness of low frequency perturbations'. In: *arXiv preprint arXiv:1903.00073* (2019).

[59] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra. 'Grad-cam: Visual explanations from deep networks via gradient-based localization'. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 618–626.

[60] K. Dabov, A. Foi, V. Katkovnik and K. Egiazarian. 'Image denoising by sparse 3-D transform-domain collaborative filtering'. In: *IEEE Transactions on image processing* 16.8 (2007), pp. 2080–2095.

[61] G. K. Wallace. 'The JPEG still picture compression standard'. In: *Communications of the ACM* 34.4 (1991), pp. 30–44.

[62] G. J. Burton and I. R. Moorhead. 'Color and spatial structure in natural scenes'. In: *Applied optics* 26.1 (1987), pp. 157–170.

[63] D. J. Tolhurst, Y. Tadmor and T. Chao. 'Amplitude spectra of natural images'. In: *Ophthalmic and Physiological Optics* 12.2 (1992), pp. 229–232.

[64] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich. *Going Deeper with Convolutions*. 2014. arXiv: 1409.4842 [cs.CV].

3

STEALTHY BACKDOOR ATTACK AGAINST FEDERATED LEARNING

Federated Learning (FL) is a beneficial decentralized learning approach for preserving the privacy of local datasets of distributed agents. However, the distributed property of FL and untrustworthy data introduce the vulnerability to backdoor attacks. In this attack scenario, an adversary manipulates its local data with a specific trigger and trains a malicious local model to implant the backdoor. During inference, the global model would misbehave for any input with the trigger to the attacker-chosen prediction. Most existing backdoor attacks against FL focus on bypassing defense mechanisms, without considering the inspection of model parameters on the server. These attacks are susceptible to detection through dynamic clustering based on model parameter similarity. Besides, current methods provide limited imperceptibility of their trigger in the spatial domain.

To address these limitations, we propose a stealthy backdoor attack called "Chironex" against FL with an imperceptible trigger in frequency space to deliver attack effectiveness, stealthiness and robustness against various countermeasures on FL. We first design a frequency trigger function to generate an imperceptible frequency trigger to evade human inspection. Then we fully exploit the attacker's advantage to enhance attack robustness by estimating benign updates and analyzing the impact of the backdoor on model parameters through a task-sensitive neuron searcher. It disguises malicious updates as benign ones by reducing the impact of backdoor neurons that greatly contribute to the backdoor task based on activation value, and encouraging them to update towards benign model parameters trained by the attacker. We conduct extensive experiments on various image classifiers with real-world datasets to provide empirical evidence that Chironex can evade the most recent robust FL aggregation

This chapter is based on the paper "Stealthy Backdoor Attack against Federated Learning through Frequency Domain by Backdoor Neuron Constraint and Model Camouflage" by Qiao, Y., Liu, D., Wang, R., Liang, K. In IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS), 2024.

algorithms, and further achieve a distinctly higher attack success rate than existing attacks, without undermining the utility of the global model.

3.1. INTRODUCTION

Federated Learning (FL) [1–8] is a type of distributed machine learning framework that has been proposed to preserve data privacy among participating agents. It supports collaborative training of an accurate global model by allowing agents to upload local updates, such as gradients and weights, to a server without compromising the local datasets. FL has been applied to various real-world applications including COVID-19 prediction [9] and autonomous driving [10].

Despite its attractive advantages, FL is susceptible to backdoor attacks [11–17]. [13] introduced a distributed backdoor attack (DBA) by dividing a global trigger into multiple pieces, which are distributed among local agents. The DBA incurs significant changes to certain dimensions of parameters in order to maintain the accuracy of the backdoor task. Recently, [12] proposed a stealthy model poisoning (SMP) attack by limiting Euclidean distance between the average updates (from all the benign agents) and the malicious updates. The malicious updates derived by this attack can be distinguished from the benign updates in the output layer at the parameter level because of the noticeable distances between them. Both DBA and SMP can be defended by using robust FL aggregation algorithms based on dynamic clustering via HDBSCAN [18] due to the significant directional discrepancy between the updates derived from these attacks and benign updates. Additionally, some works focus on developing durable backdoor attacks against FL to maintain high attack effectiveness when the adversary stops updating malicious models or gradients. For instance, Neurotoxin [19] attacks parameters that are changed less in magnitude during training which improves the durability of backdoors.

To mitigate backdoor attacks, researchers have designed robust FL aggregation algorithms [20–23]. For instance, FLAME [22] and DeepSight [23] apply HDBSCAN at the training stage to conduct clustering and filter out malicious updates. Specifically, FLAME exploits the discernible difference of model weights based on cosine similarity between benign and malicious updates, while DeepSight filters malicious updates by the output difference between benign and malicious models and the distinction of distribution of labels in the underlying training data of those models. They use clipping strategies to reduce the influence of malicious updates and beyond. Moreover, FLAME uses adaptive noise to smooth the boundary of clustering. The current approach of defense is to detect malicious updates by exploiting the distinguishable dissimilarity between updates from malicious and benign agents. Additionally, current attacks [13, 14, 16, 19] do not provide enough trigger stealthiness during the inference stage. The poisoned samples with perceptible perturbation can be easily identified by an evaluator or a user who can distinguish the difference between 'just an incorrect classification/prediction of the global model and the purposeful wrong decision due to a backdoor in the test/use stage.

There exists an impossibility for backdoor attacks to evade existing defenses because the backdoor tasks have a significant and noticeable impact on the backdoor-sensitive neurons deriving the distinguished distance from benign updates. This raises the question: *could we disguise malicious updates as benign ones at the parameter level to bypass current detection strategies while still maintaining the accuracy of the backdoor attack?*

To provide a concrete answer to the question, we propose a stealthy frequency attack, Chironex, to backdoor robust FL systems. Specifically, Chironex utilizes a frequency trigger function to produce the poisoned dataset, ensuring natural stealthiness of poisoned samples. Recent works [24–27] have provided evidence that frequency domain triggers are still learnable by neural networks and provide excellent natural stealthiness. Then, Chironex constrains the impact of backdoor neurons identified by a new method called the Task-sensitive Neuron Searcher (TNS). TNS can construct a “backdoor" neuron list consisting of neurons that deliver a significant contribution to the backdoor task so that we can penalize the weights and biases of these neurons when their updates are in malicious directions. Chironex can enforce malicious updates to be naturally imperceptible from benign ones by minimizing the distance between malicious and benign parameters owned by the attacker.

Our main **contributions** are summarized as follows. We first design a frequency trigger function to produce imperceptible poisoned samples. We analyze the behavior of neurons in backdoor tasks at the parameter level by using TNS to identify those backdoor neurons that significantly contribute to backdoor tasks and reduce the impact of backdoor-sensitive neurons. We apply a step-forward training approach to generate benign and malicious models. Furthermore, we combine the malicious model with TNS to find the list of backdoor neurons and minimize its impact in order to evade anomaly parameter detection; meanwhile, we use the benign model as an estimation of the attacker’s expected local model. We restrain parameter dissimilarity to make malicious updates indistinguishable from benign updates trained by the attacker (i.e, obtaining model camouflage) without sacrificing the utility of the global model. We fully take advantage of the attacker’s ability to provide the criterion of malicious model update direction. Finally, we evaluate the attack performance and stealthiness on real-world datasets with various datasets and models. The experimental results demonstrate that Chironex achieves a high attack success rate while maintaining global model accuracy. Our attack also provides excellent stealthiness, allowing it to bypass the most recent robust aggregation algorithms, e.g., FLAME, DeepSight, whilst other existing attacks cannot. For example, Chironex achieves around 97.60% attack success rate and 90.65% global model accuracy under FLAME on FMNIST.

The rest of the paper is organized as follows. In Section 3.2, the state-of-the-art federated learning frameworks, backdoor attack and defensive methods against federated learning are provided. Section 3.3 presents the threat model of our attack, including attack goal, capability and knowledge. Section 3.4 details the technical approach including trigger function, backdoor neuron search and model camouflage. The experimental results of attack performance and ablation studies are given in Section 3.5. Finally, the main conclusion and discussion are provided in Section 3.6.

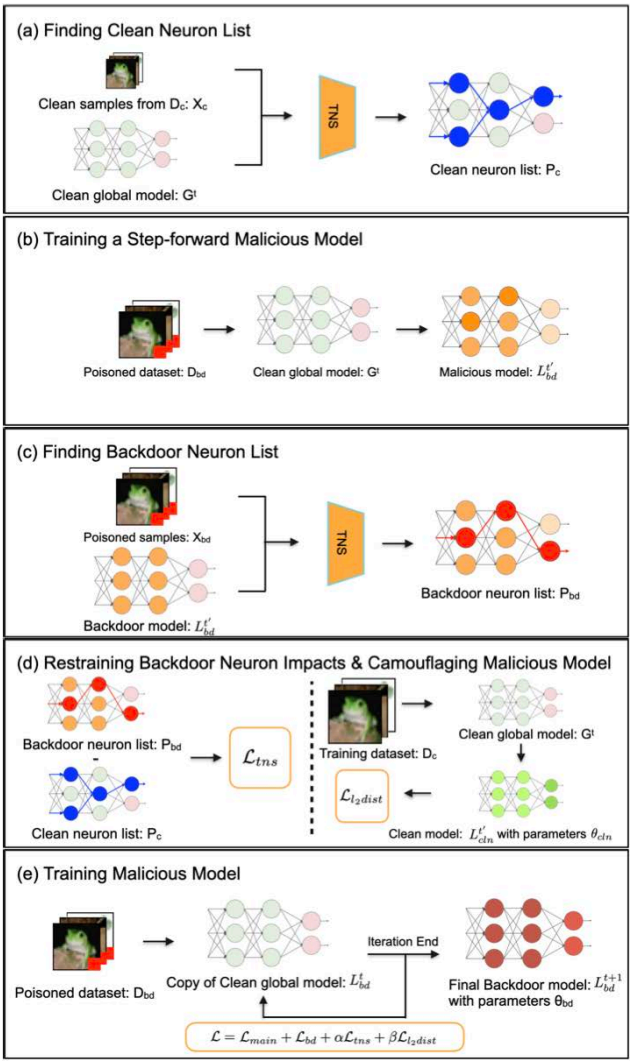


Figure 3.1: The workflow of Chironex. The Chironex attack includes three objectives: (1) achieving high accuracy on clean and backdoor tasks; (2) minimizing the impacts of backdoor neurons; and (3) minimizing l_2-norm distance between malicious parameters and estimated benign ones. At round t, we train malicious update δ_m^{t+1} as (a) Find the list P_c of clean neurons that contributed to main task by the proposed TNS; (b) Pretrain a malicious local model for searching backdoor neurons; (c) Find the list P_{bd} of backdoor neurons that contributed to backdoor task by TNS; (d) Compute the constraint loss of backdoor neurons $\mathscr{L}_{tns}$ based on P_c and P_{bd} and train θ_{cln} for $\mathscr{L}_{l_2dist}$; (e) Optimize final malicious model θ_{bd} by $\mathscr{L}$, including classification loss $\mathscr{L}_{main} + \mathscr{L}_{bd}$, backdoor constraint loss and model camouflage loss on D_{bd} to obtain malicious update δ_m^{t+1}.

3.2. RELATED WORK

3.2.1. FEDERATED LEARNING (FL)

[1] proposed the concept of distributed learning associating with n agents and a server S to train a global model G collaboratively. At the training round t, each local agent i uses the global model G^t of the current round to train a local model L_i^{t+1} based on its own data D_i and sends the parameters/gradients update δ_i to S. Then the server S aggregates the received updates $\delta_i|_{i=1}^n$ from all the agents into the global model G^t to derive G^{t+1}. In the above process, each agent i computes the update as $\delta_i^{t+1} = L_i^{t+1} - G^t$, so that S uses an aggregation algorithm to compute a new global model G^{t+1} as:

$$G^{t+1} = G^t + \frac{lr_s}{n} \sum_{i=1}^{n} (L_i^{t+1} - G^t), \tag{3.1}$$

where lr_s is the learning rate of the server. The most commonly used aggregation algorithm is FedAvg, averaging the weighted updates of all the local agents as $G^{t+1} = \sum_{i=1}^{n} \frac{d_i}{d}(L_i^{t+1} - G^t)$, where $d_i = |D_i|$, $d_i = |D_i|, d = \sum_{i=1}^{n} d_i$. We set $G^{t+1} = \sum_{i=1}^{n} \frac{1}{n}(L_i^{t+1} - G^t)$ for the equal contribution of all the clients to evade from receiving fabricated dataset size of malicious agents as in [22]. When G converges or the training reaches a specific iteration upper bound, the aggregation process terminates and outputs a final global model.

Optimizations of FL have been proposed for various purposes, e.g., privacy [28], security [29], heterogeneity [30], communication efficiency [31] and personalization issues [32].

3.2.2. BACKDOOR ATTACKS ON FL

An attacker can easily corrupt a set of agents in the training stage. These agents are manipulated to use poisoned data with a specific trigger and change an attacker-chosen base label to a target label to train their local models and further, they send the updates to the server performing aggregation. Accordingly, the global model that combines the updates from the malicious agents is embedded with a "backdoor". In the test stage, the model easily misclassifies the data inputs with the backdoor trigger to the target label. To design a successful attack, the attacker must ensure that the clean dataset is classified into the correct label and the utility of the global model cannot be harmed by backdoor task training. Note that this notorious target poisoning can seriously affect the model prediction results and is difficult to detect after the training stage. Existing backdoor attacks may target to data and models.

Data poisoning attacks. The attacker manipulates training datasets by adding backdoor patterns into data samples. For example, it disintegrates the global pattern into several local patterns and further injects them to the compromised agents' datasets separately in DBA [13]. Although making global trigger more insidious, DBA does not restrain the training process so that the malicious and benign updates can

look different at the model parameter level. Moreover, the attacker can modify the training dataset by flipping its label [33] and later send the model update trained by mislabeled data to the server. This type of attacks focuses on manipulating training data without fully considering the server aggregation strategies for anomaly updates detection.

Model poisoning attacks. This type of attacks manipulates the training process of malicious agents and further evades aggregators anomaly update detection. [14] introduced model replacement and scaling up attack to FL systems. The attacker can scale up the malicious update by a specific factor and the global model is replaced by the malicious model trained by poisoned data consequently. This attack brings a new perspective that an attacker can manipulate the training process or local model to perform the backdoor attack. [11] proposed a so-called LIE attack by crafting model updates with minor changes. The attack explores the maximized range of parameters perturbation to induce the model to predict the desired label. [15] exploited a projected gradient descent (PGD) attack with model replacement during the training of a malicious model on a dataset that is similar but not identical to the (original) clean dataset. [12] designed an attack aiming to achieve stealthiness by estimating the average update from all benign updates and reducing the L_2-norm distance between the malicious and the average updates from others. The attack, however, cannot provide a solid stealthiness because there still exists a noticeable (peak) difference in the distribution of parameters between malicious and benign updates. Some works also aim to improve the persistence of attack effectiveness on FL. For example, Neurotoxin [19] attacks parameters that are changed less in magnitude during training which improves the durability of backdoors. Chameleon [34] finds that benign images with the original and the target labels of the poisoned images have key effects on backdoor durability. It then utilizes contrastive learning to amplify such effects towards a more durable backdoor.

3.2.3. ROBUST FL AGGREGATION

Several works have been proposed to handle malicious agents in the context of FL [35–41]. Krum [42] selects a local model that is similar to others as the global model, but it is vulnerable to some dimensions of malicious model parameters. Bulyan [43] improves Krum by applying a variant of Trimmed Mean method. Trimmed Mean [20] aggregates each dimension of model parameters independently and it computes the mean for a range of parameters. Median [20] takes median for aggregation. [44] indicated that one can use FedAvg aggregation rules, by clipping weights and adding noise, to mitigate backdoor attacks. [21] proposed a robust aggregation algorithm based on sign aggregation [45] so-called RLR which changes the central server's learning rate based on the signs of agents updates. Recently, [22] proposed a defending framework based on the clustering algorithm (HDBSCAN) so-called FLAME which can cluster dynamically all local updates based on their cosine distance into two groups separately. FLAME uses weight clipping for scaling-up malicious weights and noise addition for smoothing the boundary of clustering after filtering malicious updates. [23] designed a robust FL aggregation rule called DeepSight using HDBSCAN. Their design leverages parameter distribution, output, and cosine

distance to cluster all updates and further applies the clipping method. DeepSight fully exploits information leakage from malicious updates and provides a more precise detection than FLAME. SparseFed [46] performs norm clipping to all local updates and averages the updates as the aggregate. Top-k values of the aggregation update are extracted to filter out potential malicious parameters and returned to each agent who locally updates the models using this sparse update. CRFL [39] provides certified robustness in FL frameworks. It exploits parameter clipping and perturbing during federated averaging aggregation. In the test stage, it constructs a "smoothed" classifier using parameter smoothing. The robust accuracy of each test sample can be certified by this classifier when the number of compromised clients or perturbation to the test input is below a certified threshold.

3.3. THREAT MODEL AND MOTIVATION

We consider the same threat model as in [12–14].

3.3.1. ATTACKERS GOAL

Following [13–15, 19], we enable the attacker to manipulate the global model to predict a target label on any samples with an attacker-chosen trigger (i.e. the backdoor task). From the viewpoint of the attacker, there are two main objectives: preserving functionality for benign tasks (i.e., maintaining global model accuracy) and ensuring attack effectiveness for backdoor tasks (i.e., achieving high backdoor accuracy on the poisoned model). Two additional goals are considered in our attack including attack stealthiness and robustness. Attack stealthiness implies that poisoned samples exhibit visual similarity to clean ones while robustness is demonstrated by its effectiveness against backdoor defenses on FL. Unlike untargeted poisoning attacks [47] preventing the convergence of the global model, the goal of our attack is to manipulate malicious agents' local training processes to achieve high accuracy in the backdoor task without undermining the main task. Thus, the global model's behavior is naturally normal on clean data samples while it will predict poisoned data into a target label with a high attack success rate.

3.3.2. ATTACKER'S CAPABILITY AND KNOWLEDGE

We assume the attacker only has full access to malicious agents' clients, local training processes and training datasets. The attacker cannot change the aggregation algorithm of the server and manipulate the training processes and training datasets of the honest agents so that the model updates of those agents will not be affected by the attack. Unlike some backdoor attacks strictly requiring malicious agents to collude together, our assumption here does not need this collusion. Note we naturally allow malicious agents to collude but our attack can work well without this collusion, which means that we require fewer restrictions than others. At last, we do not require the attacker to know the FL aggregation rules applied in the server.

3.3.3. TECHNICAL MOTIVATION

Existing attacks cannot reduce the distance between benign and malicious update in the parameter space and thus they can be easily detected by the state-of-the-art robust FL systems using parameter inspection. We study a new attack perspective in the sense that the attacker can restrain the disparity among parameters by training loss function of l_2-norm distance or cosine similarity. Some attacks, such as SMP, have used the similar objective terms to constrain the distance. In Figure 3.3, we show that it is not sufficient yet to only limit the distance among the parameters as we can still see the difference between benign and malicious updates. To tackle the issue, we introduce a new objective to further eliminate the influence of those backdoor neurons found by TNS. Unlike SMP, the philosophy of Chironex is to convert the malicious parameters to become benign, i.e. fooling the server to regard malicious updates as benign ones. In the figure, we see that the malicious model parameters of Chironex are very close to the benign parameters so that the malicious updates are much less abnormal. We further state that the attacker does not need to know a concrete FL aggregation algorithm before applying our attack, which makes the attack more general and practical. The experimental results (see Section 3.5.4) show that two penalties $\mathscr{L}_{tns}$ and $\mathscr{L}_{l_2dist}$ work well with the combination of classification loss $\mathscr{L}_{main}$ and $\mathscr{L}_{bd}$ so that we can eliminate the impact of those backdoor neurons and meanwhile achieve practical performance w.r.t. attack success rate (ASR) and global model accuracy (MA).

3.4. PROPOSED METHODOLOGY

We provide problem formulation and technical details in this section. Specifically, we first introduce our trigger injection function to insert a specific pattern into the frequency domain. Then, we formulate the optimization problem by three attack objectives, i.e., (1) high accuracy on clean and backdoor tasks, (2) backdoor neuron constraint and (3) model camouflage. We utilize step-forward training to obtain benign and malicious reference models on clean and poisoned datasets. We introduce a novel method to search backdoor neurons on the malicious reference model and constrain the impact of those neurons. Additionally, we minimize the l_2-norm distance between malicious parameters and benign ones from the benign reference model to achieve model camouflage. The frequently used notations in this paper are shown in Table 3.1.

3.4.1. PROBLEM FORMULATION

Trigger injection function ϕ. Taking the image classification task as an example, we first introduce a frequency trigger function $\phi(\cdot)$ to produce the poisoned dataset D_{bd} with the imperceptible trigger t through frequency space. Given a clean sample $x \in [0,1]^{H\times W\times C}$ (height H, width W and channel C) and a specific trigger image x^t, we first transform them into frequency domain via discrete cosine transform (DCT) $\mathscr{D}(\cdot)$ as:

$$\mathscr{D}(u,v,c) = N_u N_v \sum_{i=0}^{H-1}\sum_{j=0}^{W-1} x(i,j,c) cos\frac{(2i+1)u\pi}{2H} cos\frac{(2j+1)v\pi}{2W}, \tag{3.2}$$

Table 3.1: Notation Summary.

Notation	Description	Notation	Description
δ^t	Update of agent at round t	P	Neuron list of TNS
D_c	Clean dataset	α, β	Lagrange coefficients
D_{bd}	Poisoned dataset	G^t	Global model at round t
ω	Blend ratio	y_t	Target label
$\phi(\cdot)$	Trigger injection function	ξ	Threshold for P
η	Malicious agents rate	lr	Learning rate
L_i^t	Local model of agent i at round t	ρ	Threshold of RLR
γ	Neuron sensitivity rate	S	FL Server
$a_{[i]}^l$	Activation of neuron i of layer l	$\psi(\cdot)$	Measurement of P
$b_{[i]}^l$	Bias of neuron i	s	Cluster size of Defenses
$n_{[i]}^l$	Neuron i	X	# of Samples in TNS
$w_{[i]}^l$	Weight of neuron i	n	# of total agents
m	# of malicious agents	R	Total training round

where $u, i \in \{0, 1, \cdots, H-1\}$, $v, j \in \{0, 1, \cdots, W-1\}$ and $c \in \{0, 1, \cdots, C-1\}$. N_u and N_v are normalization terms, $N_u \triangleq \sqrt{1/H}$ if $u = 0$ and otherwise $N_u \triangleq \sqrt{2/H}$. Similarly, $N_v \triangleq \sqrt{1/W}$ if $v = 0$ and otherwise $N_v \triangleq \sqrt{2/W}$. Triples (i, j, c) and (u, v, c) refer to a specific pixel and frequency band of x and its frequency form respectively. Then, we blend the trigger pattern t (spectrum of $\mathscr{D}(x^t)$) and $\mathscr{D}(x)$ to generate the poisoned sample x_{bd} with a binary mask $\mathscr{M} = 1_{(u,v,c) \in [0:\lambda H:, 0:\lambda W:]}$, where λ determines the location and size of trigger to be blended and ω is the blend ratio to decide the proportion of information contributed by x^t. Finally, we utilize inverse DCT (IDCT) $\mathscr{D}^{-1}(\cdot)$ to obtain the spatial form of poisoned sample x_{bd}. The entire frequency trigger injection method is held:

$$\phi(x) = \mathscr{D}^{-1}(\mathscr{D}(x)(1 - \mathscr{M}) + [(1 - \omega)\mathscr{D}(x) + \omega t] * \mathscr{M}). \tag{3.3}$$

The goal of this function is to mislead the prediction of $\phi(x)$ to the target label y_t.

Attack objectives. We propose a stealthy backdoor attack against FL on computer vision tasks. In the following, we denote the clean training dataset as $D_c = \{(x_i, y_i)\}_{i=1}^{|D_c|}$ containing $|D_c|$ images. In practice, we randomly select samples from D_c to produce poisoned training dataset D_{bd} by the proposed $\phi(\cdot)$ with a specific poison ratio. For a clean sample and its label (x, y) from D_c, we poison the clean sample to $(\phi(x), y_t)$ by backdoor injection function $\phi(\cdot)$, where y_t is the target label.

Our main objective is to learn backdoor parameters θ_{bd} (trained by D_{bd}) by constraining the influence of backdoor neurons and making the parameters close to the benign parameters θ_{cln} trained a step-forward by D_c. Given a malicious local model L_{bd}^t at round t with θ_{bd} and a backdoor injection function $\phi(\cdot)$, we minimize the following loss function to hold the performance of the main task on D_c:

$$\mathscr{L}_{main} = \sum_{(x,y) \in D_c} \mathscr{L}^{ce}(L_{bd}^t(x), y), \tag{3.4}$$

where $\mathscr{L}^{ce}$ denotes the cross entropy loss. To provide a practical attack success rate, we minimize the following loss function on the backdoor task:

$$\mathscr{L}_{bd} = \sum_{(x,y)\in D_{bd}} \mathscr{L}^{ce}(L^t_{bd}(\phi(x)), y_t). \tag{3.5}$$

We notice that the backdoor training is a kind of “shortcut" learning which means several backdoor-sensitive neurons are easily affected via the backdoor task with certain preferences while they have less contribution on the main task training. To achieve stealthiness, we minimize the following constrained loss function on poisoned data to reduce the impact of those backdoor-sensitive neurons:

$$\mathscr{L}_{tns} = \sum_{(i,l)\in\psi(P_{bd},P_c)} (|w^l_{[i]}| + |b^l_{[i]}|), \tag{3.6}$$

where $\mathscr{L}_{tns}$ is backdoor neurons constraint loss, $\psi(\cdot)$ is our measurement between the backdoor neuron list P_{bd} and the clean neuron list P_c found by benign and backdoor TNS, $w^l_{[i]}$ and $b^l_{[i]}$ are the weight and bias corresponding to the i-th neuron $n^l_{[i]}$ of the l-th layer. The detailed information about how to obtain two lists P_c and P_{bd} are provided in Section 3.4.2.

To capture parameter similarity to support model camouflage, we restrain the loss corresponding to l_2-norm distance between θ_{bd} and θ_{cln}:

$$\mathscr{L}_{l_2 dist} = ||\theta_{bd} - \theta_{cln}||_2, \tag{3.7}$$

where θ_{bd} is the parameters of malicious model and θ_{cln} is the estimated benign parameters. The details of model camouflage are provided in Section 3.4.3.

Given the three objectives from eqs. (3.4) and (3.5), eq. (3.6) and eq. (3.7), we formalize the final attacker's objectives as a constrained optimization problem:

$$\arg\min_{\theta_{bd}} \mathscr{L}_{main} + \mathscr{L}_{bd} + \alpha\mathscr{L}_{tns} + \beta\mathscr{L}_{l_2 dist}, \tag{3.8}$$

where we use α and β to control the strength of the constraint loss. We can achieve the optimization by constraining the contribution introduced by backdoor neurons (which are identified by TNS) and implementing the model camouflage. The overview of Chironex is in Figure 3.1.

3.4.2. BACKDOOR NEURON CONSTRAINT BY TNS

To compute backdoor neuron constraint in Equation (3.6), we here use the proposed TNS to find task-sensitive neurons in each layer of Deep Neural Networks (DNNs). We first give the task-sensitive (influential) neurons (TSN) in Definition 3.4.1 by following the same philosophy as in [15]. The neurons satisfying the (task) sensitivity contribute significantly to a certain task[1] (i.e. either a main or a backdoor task); and if they are sensitive to a backdoor task, we call them backdoor neurons. We use the activation value of a neuron to measure its contribution or influence of a classification.

[1]These TSN deliver more contributions than others in the task. Changing their weights also outputs a crucial impact on the specific task (while this doesnt apply to the non-TSN)

Algorithm 3: Task-sensitive Neuron Searcher (TNS)

Input: Samples X, Model $\mathscr{M}$
Parameter: Number of Linear Layers L, Weights w of Model $\mathscr{M}$, Activation Value a, Neurons n, Index of Neurons i, j, Sensitivity Rate γ, Target Label y_t, Neuron List Π
Output: List P
Initialize $\Pi = []$
Initialize $P = \{0\}$
for $X_i \in X$ **do**
  $GetActivationValue(\mathscr{M}, X_i)$
  $Append(\Pi^L, n^L_{[y_t]})$
  $P[n^L_{[y_t]}] \mathrel{+}= 1$
  for $l \in L$ *(descend order)* **do**
    for $n^l_{[i]} \in \Pi^l$ **do**
      for $n^{l-1}_{[j]} \in n^{l-1}$ **do**
        if $|w^{l-1}_{[j]} * a^{l-1}_{[j]}| > |\gamma * a^l_{[i]}|$ **then**
          $Append(\Pi^{l-1}, n^{l-1}_{[j]})$
          $P[n^{l-1}_{[j]}] \mathrel{+}= 1$
        end
      end
    end
  end
end
return P

Definition 3.4.1 (Task-sensitive Neuron)**.** *Given a real positive number γ, a neuron n, its activation value a, weight w, the activation value A of the neuron of the next layer connecting to n, the neuron n for classification is task-sensitive with γ-sensitivity if* $|w \times a| > |\gamma \times A|$.

To effectively search the sensitive neurons, we apply a mixed strategy including forward and backward analysis. For the deeper linear layers, where neurons contribute more significantly compared to the shallow layers (e.g., convolutional layers), we apply backward analysis to meticulously identify task-sensitive neurons. First, we feed input samples to DNN. For a certain neuron $n^L_{[i]}$ of layer L, we compare all activation values $a^{L-1}_{[j]}|^M_{j=1}$ of M neurons $n^{L-1}_{[j]}|^M_{j=1}$ in the previous layer $L-1$, connecting to $n^L_{[i]}$, to activation value $a^L_{[i]}$ of this neuron. Second, we set a sensitivity rate γ for the current layer L. We compute each neuron's contribution as $|w^{L-1}_{[j]} * a^{L-1}_{[j]}|$, where $w^{L-1}_{[j]}$ is the weight of $n^{L-1}_{[j]}$. If $|w^{L-1}_{[j]} * a^{L-1}_{[j]}| > |\gamma * a^L_{[i]}|$, $n^{L-1}_{[j]}$ can be identified as a backdoor neuron which contributes more to the backdoor task in this layer. Algorithm 3 shows how we identify all task-sensitive neurons and generate the neuron list. For shallow layers, we use forward analysis to select the top 5% of

neurons with the highest activations in each layer. This approach enables our attack to establish connections between the layers.

We further minimize $\mathscr{L}_{tns}$ by TNS based on benign and malicious models with clean and poisoned samples to rectify those backdoor neurons. The minimization of this objective provides a crucial functionality: eliminating the distinguishable impacts on backdoor parameters. At round t, we assume the attacker chooses clean samples X_c from D_c as input of the current global model G^t. We use TNS to generate a clean neuron list of global model G^t by X_c. Then, we train a step-forward malicious model $L_{bd}^{t'}$ with D_{bd} to identify backdoor neurons to be rectified. Given G^t, $L_{bd}^{t'}$, X_c and D_{bd}, we apply TNS to find the neuron lists P_c and P_{bd} for main and backdoor tasks respectively. We use threshold ξ to figure out distinguishable backdoor neurons from clean ones. We set $\psi(P_{bd}, P_c) = \{(i,l) | (P_{bd}[n_{[i]}^l] - P_c[n_{[i]}^l]) > \xi, l \in \{1, 2, \cdots, L\}\}$. If $(i,l) \in \psi(P_{bd} - P_c)$, we add the absolute value of the difference between ϵ and $w_{[i]}^l$, $b_{[i]}^l$ of $n_{[i]}^l$ to $\mathscr{L}_{tns}$. To eliminate the influence of backdoor neurons, we enforce $w_{[i]}^l$, $b_{[i]}^l$ to approach zero. Finally, we compute TNS loss function as $\mathscr{L}_{tns} = \sum_{(i,l) \in \psi(P_{bd} - P_c)} (|w_{[i]}^l| + |b_{[i]}^l|)$ and train θ_{bd} with it. The details are shown in Figure 3.1 (a) (b) (c) and (d)-left, and in lines $2-5$ and 8 of Algorithm 4.

Why does TNS work? We found that backdoor neurons are distinguishable from the clean neurons (which mostly contribute to the main task) and meanwhile, they could only and significantly contribute to the backdoor task. By minimizing $\mathscr{L}_{tns}$ and $\mathscr{L}_{bd}$, the contribution of the backdoor neurons is redistributed to other neurons that have less contribution to the backdoor task. In this way, we encourage those neurons with "less" contribution to play a part in the backdoor task. In Figure 3.6, we show the results of the disparity between the backdoor and clean neuron lists to confirm that minimizing $\mathscr{L}_{tns}$ can reduce the contribution of those backdoor neurons at the parameter level, without affecting the sensitivity of the clean neurons in the main task. Thus, we can maintain a high attack accuracy but also have a low level of influence incurred by those backdoor neurons.

3.4.3. MODEL CAMOUFLAGE

Given the computation of $\mathscr{L}_{tns}$, our next task is to disguise θ_{bd} as θ_{cln} by minimizing the distance loss $\mathscr{L}_{l_2 dist}$ in Equation (3.7). To this end, we design a new backdoor training approach motivated by [12] to manipulate the malicious parameters more naturally like benign ones trained by D_c (which are owned by the attacker). Instead of estimating the current global model parameters through averaged updates from the previous iteration [12], we get the malicious and benign parameters sufficiently close to each other which makes the attack stealthy. We allow the attacker to train both two models based on the same set of its own data, which doesnt violate our assumptions. Similar to the backdoor neuron constraint process, we also use step-forward training with D_c to obtain a benign reference model $L_{cln}^{t'}$ to estimate the expected benign model parameters θ_{cln}. During backdoor training, we use Equation (3.7) to enforce the malicious parameters to be indistinguishable from θ_{cln} in the training stage. We set l_2-norm distance as the measurement for parameter similarity and β as a hyperparameter that controls the strength of model camouflage

between θ_{cln} and θ_{bd}. The computation of $\mathscr{L}_{l_2dist}$ is shown in Figure 3.1 (d)-right and (e), and in lines 6 and 8 of Algorithm 4. As last, we train the malicious model with D_{bd} via Equation (3.8). The details of Chironex are given in Algorithm 4.

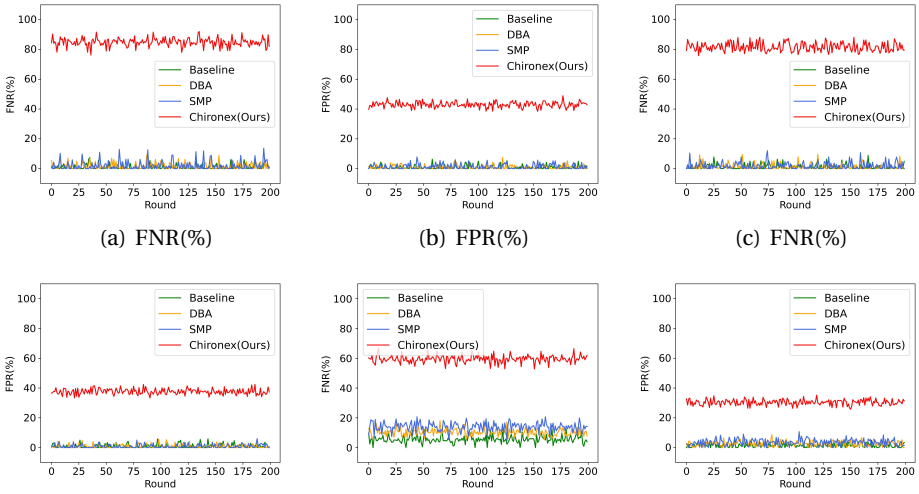
3.5. EXPERIMENTS

3.5.1. EXPERIMENTAL SETUP

Datasets and Network Structures. We tested the effectiveness of Chironex on five standard image datasets: MNIST [48], Fashion-MNIST (FMNIST) [49], CIFAR-10 [50], FEMNIST [51] and Tiny-ImageNet [52] in the independent and identically distribution (i.i.d). and non-i.i.d. data distribution settings. Specifically, MNIST contains 70k (28×28) handwritten digits images divided into ten mutually exclusive classes, in which 60k are for training and 10k for testing. FMNIST provides the same amount and size of grayscale images with ten classes. CIFAR-10 includes 60k (32×32) color images with the same number of classes, in which 50k are for training and 10k for testing. FEMNIST contains about 800k (28×28) handwritten digits and characters images divided into 62 classes provided by 3,550 users, which is commonly used in federated learning frameworks. Tiny-ImageNet has 200 classes and 100k (64×64) colored images. Each class includes 500 training samples, 50 validation and 50 test samples. The data are distributed in both i.i.d. and non-i.i.d. fashion among agents. In the non-i.i.d. context, we set a q of the dataset to evaluate the degree of non-i.i.d. level by following [38]. Since there are 10 classes, we divide the agents into 10 groups.

Algorithm 4: Chironex Backdoor Attack

1 **Input**: Clean Dataset D_c, Poisoned Dataset D_{bd}, Global Model G^t with Parameters θ, Clean Samples X_c
2 **Parameter**: Threshold ξ, Hyperparameters α, β, Client Learning Rate lr_c
3 **Output**: Malicious Update δ_m^{t+1} Copy Global model: L_{bd}^t with $\theta_{bd} \leftarrow G^t$
4 Get Clean Neuron List: $P_c = \text{TNS}(G^t, X_c)$
5 A Step-forward Training for Malicious Model: $L_{bd}^{t'} \leftarrow$ Train G^t on D_{bd}
6 Poison Clean Samples: $X_{bd} = \phi(X_c)$
7 Get Backdoor Neuron List: $P_{bd} = \text{TNS}(L_{bd'}^{t'}, X_{bd})$
8 $L_{cln}^{t'}$ with Parameters $\theta_{cln} \leftarrow$ A Step-forward Training for Benign Model with G^t on D_c
9 **for** $batch \in D_{bd}$ **do**
10 Compute the Loss: $\mathscr{L} = \mathscr{L}_{main} + \mathscr{L}_{bd} + \alpha * \mathscr{L}_{tns} + \beta * \mathscr{L}_{l_2dist}$ as in Equation (3.8)
11 Train parameters θ_{bd} of L_{bd}^t with Stochastic Gradient Descent (SGD) under Learning Rate $lr_c/2$
12 **end**
13 $\delta_m^{t+1} = \theta_{bd} - \theta$
14 **return** δ_m^{t+1}



(a) FNR(%) (b) FPR(%) (c) FNR(%)

(d) FPR(%) (e) FNR(%) (f) FPR(%)

Figure 3.2: Attack Stealthiness via FNR and FPR against FLAME for different attacks. (a)-(b): MNIST, (c)-(d): FMNIST and (e)-(f): CIFAR-10.

A training sample with label $y \in \{0,1,\cdots,9\}$ is allocated to a group (and hereafter we can call it as group y) with probability $q > 0$ and to any other groups with probability $\frac{1-q}{9}$. In the group y, each agent's training data is in i.i.d manner. $q = 0.1$ represents the dataset of local agent is i.i.d; while the degree of no-i.i.d increases with a growing q. All images in the datasets are normalized to [0,1]. We performed the experiments on commonly used DNN models including the classic CNN model for MNIST, FMNIST and FEMNIST, ResNet18 [53] for CIFAR-10 and Tiny-ImageNet.

Implementation. We set $n = 1000$ agents with $m = 10$ malicious agents to train the global model for $R = 200$ rounds with FedAvg in the i.i.d. manner as the default setting. The ratio of the compromised agents to all agents $\eta = \frac{m}{n} = 1\%$. In each round, the server randomly selects 10 clients for local model aggregation. We used learning rate $lr_c = 0.1$ for local training and $lr_s = 1$ for central server aggregation while malicious agents used malicious learning rate $lr_m = 0.05$ to perform backdoor attacks. Local models were trained by SGD optimizer. We set $\frac{|D_{bd}|}{|D_c|}$ as poison data rate (PDR), which is the fraction of injected poisoned data D_{bd} with target label in the overall clean training dataset D_c with the attacker-chosen label. We set PDR = 20% as default. We found that if $\alpha, \beta \gg 1$, the malicious model parameters are close to those of the benign model but the attack performance does not work well. If $\alpha, \beta \ll 1$, the results are the other way around. To balance the trade-off, we set $\alpha = 0.5$ and $\beta = 0.5$. We choose the "Hello Kitty" pattern in [54] as our trigger image x^t as default. Some poisoned examples are shown in Figure 3.7. For Chironex attack, we set proper ξ and γ for different network structures. ξ is strongly related to the number of samples fed into models in TNS and NN layers. For TNS, we leverage

Table 3.2: Attack Performance via MA (%) and ASR (%) for several attacks against no/different defenses on MNIST, FMNIST and CIFAR-10 in the i.i.d. context. We show the averaged results (Mean±SD) of 10 independent runs.

	Defense →	No Defense		Clipping		Median		RLR		FLAME		DeepSight	
Dataset ↓	Attack ↓	MA	ASR	MA	ASR	MA	ASR	MA	ASR	MA	ASR	MA	ASR
MNIST	No Attack	99.07±0.04	-	98.72±0.06	-	98.64±0.05	-	93.05±2.38	-	98.91±0.05	-	97.39±0.16	-
	Backdoor (Baseline)	98.89±0.09	**99.96±0.05**	98.39±0.03	0.00±0.00	**98.56±0.07**	0.00±0.00	**94.95±0.23**	0.00±0.00	**98.96±0.02**	0.00±0.00	**98.36±0.05**	0.00±0.00
	DBA	98.87±0.06	98.99±0.05	98.41±0.17	0.00±0.00	98.04±0.12	4.01±3.29	93.29±2.86	0.90±1.19	98.80±0.19	1.01±0.57	97.28±0.10	0.00±0.00
	Edge-Case	98.76±0.11	98.91±0.48	**98.73±0.05**	**98.19±0.21**	98.25±0.31	0.00±0.00	93.56±2.46	0.00±0.00	98.26±0.06	0.00±0.00	97.82±0.04	0.00±0.00
	SMP	98.93±0.05	99.16±0.08	98.53±0.07	97.03±0.23	98.49±0.09	80.19±2.83	94.17±1.16	0.00±0.00	98.71±0.11	0.00±0.00	97.90±0.03	0.00±0.00
	Chironex (Ours)	**98.96±0.03**	99.38±0.62	98.17±0.04	97.93±0.17	98.53±0.08	**99.52±1.04**	94.72±1.13	**98.88±0.09**	98.84±0.07	**93.72±4.26**	97.03±0.10	**94.28±1.26**
FMNIST	No Attack	92.41±0.20	-	92.20±0.75	-	90.84±0.19	-	81.54±0.93	-	91.35±0.13	-	91.97±0.16	-
	Backdoor (Baseline)	91.05±0.04	**99.97±0.05**	**91.17±0.09**	00.00±0.00	90.42±0.07	0.00±0.00	78.77±0.17	0.00±0.00	90.04±0.06	0.00±0.00	90.63±0.20	0.00±0.00
	DBA	90.28±1.19	98.40±0.12	90.85±0.20	00.00±0.00	90.49±0.17	2.05±1.05	81.31±0.70	10.5±6.85	90.15±0.35	3.16±2.00	**91.45±0.09**	0.00±0.00
	Neurotoxin	91.22±0.18	99.83±0.04	91.05±0.36	0.00±0.00	90.39±0.15	3.26±0.91	81.20±0.62	0.00±0.00	90.49±0.58	0.00±0.00	91.03±0.07	2.41±1.78
	SMP	90.95±0.32	99.07±0.39	90.83±0.45	97.31±0.89	90.36±0.04	75.74±2.60	80.27±0.39	0.00±0.00	90.51±0.26	0.00±0.00	90.89±0.18	0.00±0.00
	Chironex (Ours)	**91.44±0.14**	97.83±0.34	90.62±0.71	**98.13±0.33**	**90.79±0.12**	**93.46±1.32**	**81.86±0.20**	**99.80±0.07**	**90.65±0.22**	**97.60±0.67**	91.29±0.13	**95.03±0.80**
CIFAR-10	No Attack	89.19±1.05	-	89.42±0.77	-	89.56±0.89	-	89.86±1.92	-	91.03±0.51	-	89.43±0.80	-
	Backdoor (Baseline)	86.35±0.67	96.95±1.79	**88.83±0.60**	0.00±0.00	**88.03±0.82**	9.72±5.15	89.07±2.10	0.00±0.00	89.20±1.29	8.40±2.29	88.14±0.82	5.07±2.38
	DBA	86.45±0.63	93.36±5.44	87.92±0.19	0.00±0.00	87.24±0.80	42.63±7.35	79.25±2.14	51.53±4.81	53.02±0.83	11.63±3.79	89.17±0.54	32.85±4.98
	Edge-Case	**86.65±0.78**	90.53±1.21	88.27±0.84	98.01±0.58	87.19±0.39	20.83±4.69	**89.33±1.75**	32.26±5.21	82.60±1.30	17.91±3.43	**89.38±0.59**	20.09±3.20
	SMP	86.30±0.64	95.47±1.02	87.09±0.57	97.04±0.99	87.27±0.69	74.86±4.46	89.04±2.05	80.53±6.17	82.48±0.59	15.88±2.94	89.26±0.87	28.27±5.05
	Chironex (Ours)	86.23±0.54	**98.52±1.61**	88.14±0.52	**98.26±0.80**	87.10±0.83	**93.37±3.29**	89.17±2.01	**95.69±4.14**	**89.24±0.76**	**91.52±3.99**	89.06±0.79	**97.97±3.71**
Tiny-ImageNet	No Attack	57.36±0.07	-	56.27±0.20	-	56.03±0.18	-	56.83±1.27	-	57.13±0.25	-	56.92±0.47	-
	Backdoor (Baseline)	56.29±0.27	**98.96±0.15**	55.39±0.31	0.00±0.00	56.73±0.91	0.00±0.00	56.39±0.73	0.00±0.00	55.07±0.76	0.00±0.00	**57.88±0.50**	0.00±0.00
	DBA	55.28±0.42	98.29±0.57	**57.31±0.63**	0.00±0.00	56.04±0.16	0.00±0.00	**57.29±1.70**	0.00±0.00	56.19±0.33	0.00±0.00	57.82±0.37	0.00±0.00
	Neurotoxin	56.82±0.12	98.78±0.06	56.67±0.17	59.98±0.40	56.71±0.34	10.48±1.95	55.85±0.96	0.00±0.00	56.83±0.39	0.00±0.00	57.14±0.67	0.81±0.33
	SMP	**57.06±0.05**	97.26±0.09	55.81±0.13	96.93±0.23	56.27±0.66	90.47±1.79	54.19±0.92	0.00±0.00	55.61±0.21	0.00±0.00	56.97±0.07	0.00±0.00
	Chironex (Ours)	56.91±0.29	98.37±0.48	56.17±0.14	**97.85±0.68**	**57.43±0.29**	**98.72±0.94**	55.21±1.03	**96.52±0.27**	**57.19±0.96**	**94.08±2.04**	57.03±0.34	**95.71±0.83**

Table 3.3: Attack Performance via MA (%) and ASR (%) for different proportion of compromised agents η (%).

	Defense →	No Defense		Median		RLR		FLAME		DeepSight	
Dataset ↓	η ↓	MA	ASR	MA	ASR	MA	ASR	MA	ASR	MA	ASR
MNIST	0.1	99.23±0.03	52.63±0.72	98.70±0.12	49.36±1.35	95.85±0.98	50.91±1.01	99.17±0.10	44.74±3.67	98.46±0.28	46.59±2.16
	0.5	99.14±0.02	98.19±0.86	98.62±0.09	97.99±1.74	94.72±0.44	97.68±0.89	99.04±0.10	91.13±4.97	98.28±0.39	92.66±2.27
	1	99.09±0.08	99.02±1.19	98.59±0.20	98.75±1.21	95.33±1.08	98.83±1.17	98.85±0.05	91.89±3.78	98.16±0.15	93.04±1.92
	5	99.03±0.04	99.25±0.79	98.50±0.05	99.10±1.63	95.96±1.93	98.29±0.15	98.89±0.09	92.53±4.56	97.81±0.26	93.72±1.77
	10	98.96±0.03	99.38±0.62	98.53±0.08	99.52±1.04	94.72±1.13	98.88±0.09	98.84±0.07	93.72±4.26	97.03±0.10	94.28±1.26
FMNIST	0.1	92.59±0.10	46.81±0.47	90.95±0.28	39.92±0.36	83.25±0.65	43.27±0.54	91.21±0.23	45.01±0.77	92.29±0.83	40.24±0.90
	0.5	92.38±0.27	96.19±1.49	90.85±0.26	90.65±0.83	82.31±0.24	98.95±0.14	91.20±0.16	95.24±0.22	91.77±0.37	93.79±1.70
	1	91.93±0.36	96.45±0.88	90.46±0.14	91.80±0.71	82.05±0.51	99.07±0.98	91.81±0.37	96.36±0.47	91.71±0.52	94.52±1.03
	5	91.46±0.85	97.80±0.30	90.35±0.82	92.20±0.40	81.92±0.30	99.34±0.56	91.24±0.35	97.68±1.10	91.90±0.15	94.77±1.15
	10	91.44±0.14	97.83±0.34	90.79±0.12	93.46±1.32	81.86±0.20	99.80±0.07	90.65±0.22	97.60±0.67	91.29±0.13	95.03±0.80
CIFAR-10	0.1	90.43±0.22	35.06±1.55	90.03±0.73	33.96±2.29	80.39±2.95	35.68±3.50	91.07±0.66	31.24±2.32	90.71±0.41	20.37±5.74
	0.5	90.41±0.64	93.48±2.11	90.55±0.83	85.79±2.66	89.81±2.86	95.40±3.21	90.12±0.92	97.63±3.79	91.56±0.09	98.37±4.25
	1	89.73±0.85	93.76±0.81	88.11±0.36	90.05±2.43	89.84±0.77	93.80±3.66	89.28±0.42	93.53±3.55	89.70±0.94	96.26±3.73
	5	87.47±0.09	96.94±1.58	87.84±0.86	92.97±2.34	88.91±1.73	84.21±3.76	85.69±0.46	97.93±3.64	88.25±0.22	98.10±3.76
	10	86.23±0.54	98.52±1.61	87.10±0.83	93.37±3.29	89.17±2.01	95.69±4.14	83.24±0.76	95.52±3.99	89.06±0.79	98.97±3.71

the entire poisoned dataset to find backdoor neurons and the same size of clean dataset to find benign ones. We set ξ as half of the number of the poisoned samples for each dataset, indicating a neuron has a significant impact on more than half of classification tasks. We set γ to 0.05 to distinguish backdoor neurons from clean ones under a given ξ, indicating the settings for ξ, γ are practical.

Evaluation Metrics:

• *Global Model Accuracy (MA).* We set the test accuracy on clean validation samples fed into the global model as MA.

• *Attack Success Rate (ASR).* We set the ratio of backdoored examples fed into the global model misclassied as the target label as ASR.

• *False Negatives Rate (FNR).* FNR evaluates the attack robustness, i.e. how well the attack evades the detection of robust FL aggregation. We set it as $\frac{FN}{FN+TP}$ indicating the ratio of malicious updates for which the defense produces wrong predictions to the total number of malicious models, i.e., the fraction of the number of malicious updates misclassified into benign updates cluster (False Negative - FN), where TP is True Positive showing the number of malicious updates correctly classified as malicious.

• *False Positives Rate (FPR).* This investigates the robustness. FPR = $\frac{FP}{FP+TN}$ denotes the ratio of benign updates that are misclassified as malicious (False Positive - FP) to the total number of benign models, where TN is True Negative indicating the number of benign updates correctly classified as benign.

3.5.2. EVALUATION OF ATTACK WITHOUT DEFENSE

We used a backdoor attack with both main and backdoor objectives as our baseline and included other attacks - DBA, Edge-Case [15], SMP [12] and Neurotoxin [19] - into the comparison. We tested our attack effectiveness through MA and ASR under FedAvg without defense. We conducted a single-target attack for each backdoor method, in which we chose a base label (class 5) misclassified into one target label y_t (class 7) for all the compared methods per dataset. The results in the i.i.d. setting are presented in Table 3.2. As for FedAvg with “no defense" on MNIST and FMNIST, Chironex achieves >98% ASR and meanwhile, it maintains MA close to benign model accuracy (< 1% gap). Although MA on CIFAR-10 is slightly lower than on other datasets, Chironex still outperforms other attacks and yields the highest ASR, roughly 95.4%. Chironex also outputs MA that is significantly close to the baseline attack, with a 0.27% gap. For a more complex dataset Tiny-ImageNet, Chironex provides excellent performance on stealthiness against SOTA defenses. Although its MA values are lower than others', Chironex is still more stealthy and delivers nearly the best ASR, >95%. As compared to its ASR results on other datasets, there is a slight drop on Tiny-ImageNet, which is roughly 1%~2%.

3.5.3. EVALUATION OF ATTACK WITH DEFENSES

Besides the norm clipping defense [55], we evaluated attack effectiveness against four robust FL aggregation rules, namely Median, RLR, FLAME, and DeepSight. For robust FL, we used RLR threshold $\rho = 40$ for each setting and set FLAME and DeepSight

minimum cluster size $s = \frac{10}{2} + 1 = 6$. The results in the i.i.d. setting are shown in Table 3.2. Since Edge-Case is not applicable on FMNIST and Tiny-ImageNet, its results are not given in Table 3.2. Chironex can provide excellent robustness against secure FL aggregation algorithms. Under FLAME and DeepSight on MNIST and FMNIST, Chironex is more robust than (most of) others and maintains high ASR values, >93%. Whilst the MA values on CIFAR-10 are slightly lower than those of other attacks, Chironex still delivers a solid ASR >93% under Median and RLR. Under DeepSight on CIFAR-10, Chironex achieves nearly 2-3× improvement on ASR as compared to others. We see that Chironex distinctively outperforms others against FLAME and DeepSight.

Evaluation of Attack with Defenses on Tiny-ImageNet. We evaluated the stealthiness of Chironex and other backdoor attacks on Tiny-ImageNet (which is a real-world and more complex dataset). In Table 3.4, Chironex provides excellent performance on stealthiness against SOTA defenses. Although its MA values are lower than others', Chironex is still more stealthy and delivers nearly the best ASR, >95%. As compared to its ASR results on other datasets, there is a slight drop on Tiny-ImageNet, which is roughly 1%~2%.

Table 3.4: Attack Performance via MA (%) and ASR (%) for several attacks against no/different defenses on Tiny-ImageNet in the i.i.d. context.

Defense	No Defense		Clipping		Median		RLR		FLAME		DeepSight	
Attack	MA	ASR	MA	ASR	MA	ASR	MA	ASR	MA	ASR	MA	ASR
No Attack	57.36	-	56.27	-	56.03	-	56.83	-	57.13	-	56.92	-
Baseline	56.29	**98.96**	55.39	0.00	56.73	0.00	56.39	0.00	55.07	0.00	**57.88**	0.00
DBA	55.28	98.29	**57.31**	0.00	56.04	0.00	**57.29**	0.00	56.19	0.00	57.82	0.00
SMP	**57.06**	97.26	55.81	96.93	56.27	90.47	54.19	0.00	55.61	0.00	56.97	0.00
Chironex	56.91	98.37	56.17	**97.85**	**57.43**	**98.72**	55.21	**96.52**	**57.19**	**94.08**	57.03	**95.71**

The FNR/FPR results against FLAME are shown in Figure 3.2. Chironex achieves an overwhelming advantage on attack effectiveness under FLAME. Its FNR can reach approx. 80% on MNIST and FMNIST while still standing at nearly 60% on CIFAR-10, which is nearly 4-8× higher than others. This indicates that FLAME could be more likely to cluster most of malicious updates into benign ones under Chironex than other attacks. As for FPR, Chironex maintains around 40% but others are seriously restricted under 10%.

To further verify the stealthiness of Chironex, we tested principal component analysis (PCA) of model parameter updates between benign and malicious agents under FLAME, see Figure 3.3. We reduced model parameters to three dimensions and compared the differences. The model update of baseline attack is distinguishable from those benign updates while malicious updates given by Chironex can be mis-identified as "benign" (where the low dimensional malicious parameters stay extremely close to benign ones). This is so because: 1) we constrain the impact of those backdoor neurons to make them be "seen" as clean neurons; and 2) we manipulate the malicious parameters to get close to the benign ones trained by the attacker. These reduce the differences between benign and malicious updates and

thus make Chironex stealthy.

Why does Chironex achieve distinctive effectiveness under robust FL? Due to our design at the parameter level, malicious and benign updates are indistinguishable.

• Against Median. Since the malicious and benign parameters are close to each other, the probability of being selected from all the updates is increased as compared to other attacks. Thus, Chironex can evade Median.

• Against RLR. The malicious updates generated by Chironex have almost the same sign of the parameters at each dimension because of the parameter similarity. Since the sum of parameter sign of our method is smaller than its threshold ρ, RLR can easily treat malicious updates as benign.

• Against FLAME and DeepSight. SMP obtains a lower ASR on CIFAR-10 than Chironex as its core design is to limit the distance between updates based on the global model of the previous round. By restraining the distance directly between malicious and benign parameters from the attacker, one may yield a precise estimation of the update at the current round. But this approach can be still identified by investigating the difference at the output layer between benign and malicious models. To bypass the detection, we focus on limiting the impact of backdoor-sensitive neurons. By introducing two loss function terms $\mathscr{L}_{tns}$ and $\mathscr{L}_{l_2dist}$, we guarantee that malicious parameters have no distinct contributions, allowing our attack to provide excellent stealthiness.

3.5.4. HYPERPARAMETER ANALYSIS

Influence of PDR and the proportion of compromised agents η. We examined our attack effectiveness and robustness under different PDR and η settings. We set PDR = 20% and $\eta = 1\%$ as default. In Table 3.5, we give the performance with FedAvg on three datasets. Chironex works very well with PDR = 50%, reaching above 98.80% (MA) and 99.55% (ASR) on MNIST and 91.30% (MA) and 98.68% (ASR) on FMNIST. Although the performance on Tiny-ImageNet is worse than those of MNIST and FMNIST, Chironex still can provide around 59.12% (MA) and 85.31% (ASR) even when PDR = 1%. We note decreasing PDR could improve MA but weaken ASR. We also investigated the performance with different η. In Table 3.3, we see that using a small η can increase MA but slightly harm ASR against defenses. Chironex presents a relatively weak performance on ASR (< 50%) when the adversary controls only one malicious client.

Influence of α, β. We used different hyperparameter settings to visualize the influence of stealthiness budget α, β on various datasets. The α, β limit the impacts of backdoor neurons and the similarity between benign and malicious updates respectively. We restrained that $\alpha, \beta \leq 1$ considering the impact of the main task and backdoor task training. The results against FLAME are in Figure 3.4. For FMNIST, reducing α (from 0.5 to 0.1) with a fixed β can make our malicious updates less indistinguishable because the contributions of backdoor neurons do increase. Reversely, fixing an α with decreasing β could perform a moderate decline on stealthiness. If we keep reducing either of them, Chironex suffers from a significant drop on ASR. For example, ASR is only around 20% with $\alpha = 0.0$ and fully declines to 0% with $\beta = 0.0$. Whilst $\alpha, \beta = 0.5$, Chironex can achieve the best stealthiness against

FLAME, with almost 100% ASR. The similar experimental results can be observed on Tiny-ImageNet.

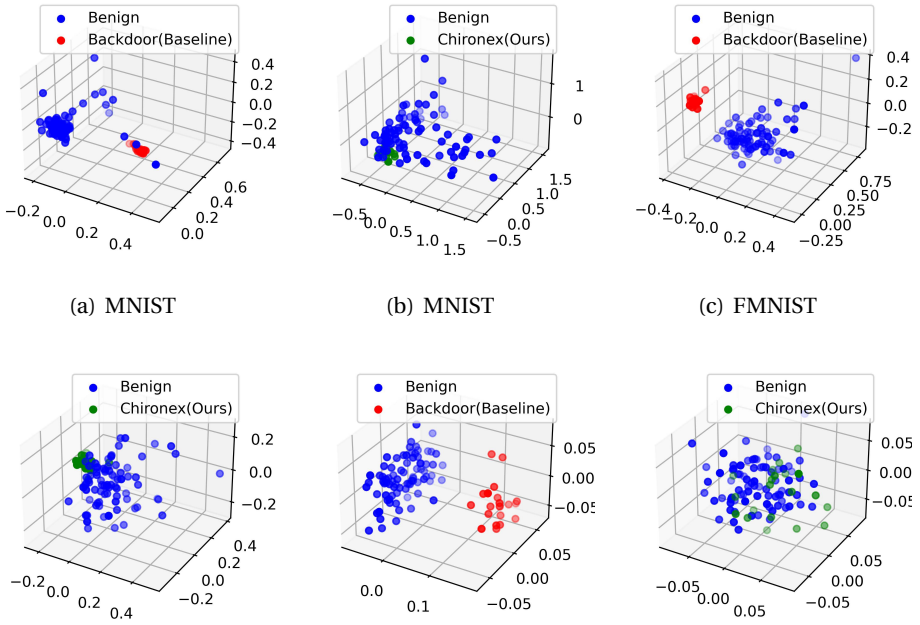
Table 3.5: Attack Performance via MA (%) and ASR (%) with FedAvg for different PDR settings.

Dataset	PDR	MA	ASR
MNIST	1%	99.13±0.14	97.14±0.43
	10%	99.27±0.28	97.21±0.23
	20%	99.03±0.13	98.46±0.07
	50%	98.80±0.08	99.55±0.11
FMNIST	1%	92.34±0.57	95.64±0.95
	10%	91.62±0.59	96.37±0.71
	20%	91.67±0.45	97.90±0.18
	50%	91.30±0.11	98.68±0.87
Tiny-ImageNet	1%	59.12±0.20	85.31±1.38
	10%	58.79±0.39	87.79±1.02
	20%	57.53±0.78	91.12±1.78
	50%	56.36±0.66	92.34±1.90

Influence of the degree of non-i.i.d. level q. We set several degrees of FEMNIST and FMNIST data distribution q to test Chironex in the non-i.i.d context. In Figure 3.5, we present its performance with robust FL aggregation rules as compared to baseline backdoor attacks on FEMNIST. From $q = 0.1$ to $q = 0.9$, Chironex achieves high ASR (>90%) and maintains MA with a small descent (<5%). But under RLR, it has a continuous drop on ASR, obtaining around 75% with $q = 0.9$. While $q = 1$, both MA and ASR experience a sharp decline (around 30% on ASR and > 50% on MA), against Median and RLR. This is so because the malicious agents cannot maintain the similarity between malicious and benign updates, which makes the attack easily detectable by the defenses. However, we see that Chironex performs more stable with $q = 1$ under FLAME and DeepSight (both <20% decrease on MA and ASR).
The disparity of Neuron lists P_{bd} and P_c. In Figure 3.6, we demonstrate the distinction in ascending order between the backdoor and clean neuron lists by subtracting approach. The results show that certain backdoor neurons of each layer do contribute to the backdoor task noticeably and significantly but they deliver no influence on the main task.

3.5.5. TRIGGER VISUALIZATION

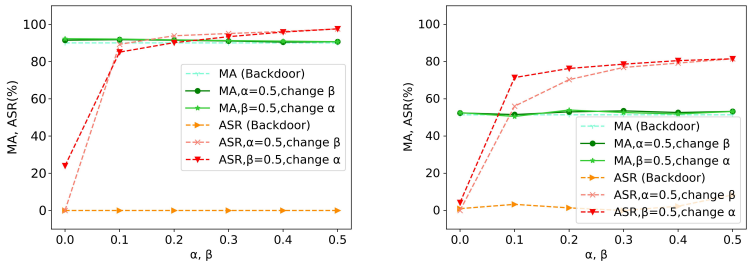
Although the attacker can arbitrarily choose the trigger because the local dataset is not visible to the server in the context of FL, using a more stealthy trigger naturally decreases the cosine dissimilarity between benign and malicious parameters [13], which could make our malicious updates more robust against defenses. To verify the natural stealthiness of the designed trigger under human inspection, we showcase poisoned samples on CIFAR-10 via our frequency trigger function in Figure 3.7. The



(a) MNIST (b) MNIST (c) FMNIST

(d) FMNIST (e) CIFAR-10 (f) CIFAR-10

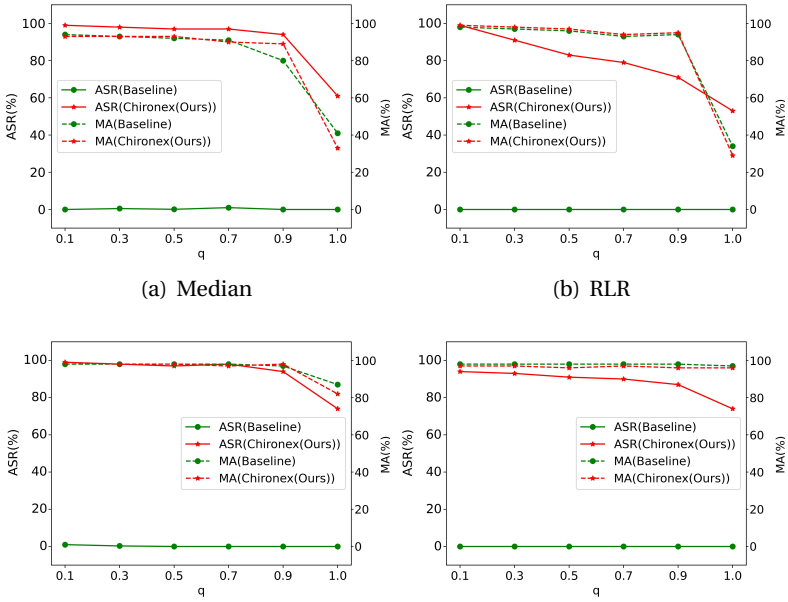
Figure 3.3: Principal component analysis (PCA) of model parameter updates for benign and malicious agents under FLAME.



(a) FMNIST (b) Tiny-ImageNet

Figure 3.4: MA (%) and ASR (%) of Backdoor (Baseline) and Chironex (Ours) with FLAME for different α, β settings.

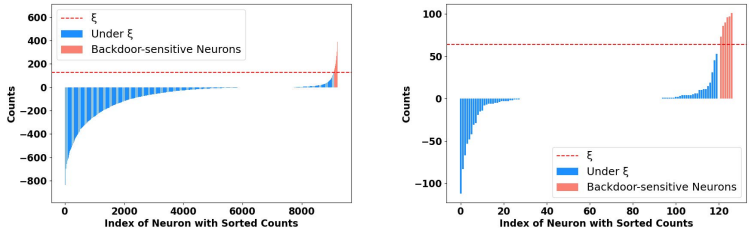
results confirm that Chironex achieves sufficient natural stealthiness that can evade human inspection.



(a) Median (b) RLR

(c) FLAME (d) DeepSight

Figure 3.5: Attack Performance via MA (%) and ASR (%) for different degrees of non-i.i.d. setting on FEMNIST. We compare Chironex (Ours) to Baseline on different defenses.



(a) FC-1 (b) FC-2

Figure 3.6: The differences between P_{bd} and P_c of each layer in ascend order by TNS for classic CNN architecture (including 2 hidden dense layers) on FMNIST.

3.6. CONCLUSION AND DISCUSSION

We designed an effective and stealthy backdoor attack throught frequency domain against FL by constraining the influence of backdoor neurons and enforcing backdoor parameters to update towards benign parameters. The empirical experiments show

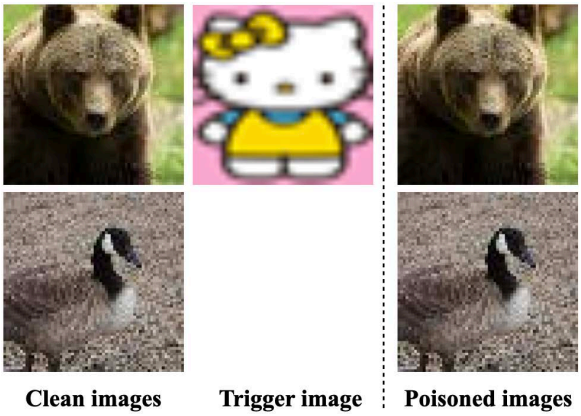


Figure 3.7: Visualization of poisoned images on Tiny-ImageNet. Our attack provides practical natural stealthiness of poisoned samples.

that our design can achieve a practical attack performance and evade most of the current defending strategies and human inspection. We hope this work could inspire further studies in developing secure and robust FL aggregation algorithms.

3.6.1. DISCUSSION.

Tasks. In this work, we concentrate on various computer vision tasks, which have been the focus of numerous existing works [13, 17]. In the future, we intend to expand the scope of this work to other machine learning tasks, e.g., natural language processing. Chironex requires additional computational costs as we apply the step-forward approach.

Dataset. Applying Chironex to more complex real-world datasets (e.g., ImageNet [56]) will not harm its attack effectiveness and stealthiness because it manipulates malicious parameters by limiting the impact of backdoor neurons rather than input samples. Beyond that, we can easily apply TNS to state-of-the-art CNN architectures, such as VGG [57], ResNet [53], which have similar structures (conv + fc) as ours. We also state that Chironex can provide attack effectiveness and stealthiness on text and speech datasets (e.g., Reddit and Sentiment140 in [51]), where the next-word prediction task and the speech recognition may use the Recurrent Neural Networks (RNNs). The proposed TNS is still applicable to RNNs. One may try to use other types of datasets and NNs at the training stage, but this will not affect the main conclusions on our attack performance.

Computational cost. Chironex requires extra computational cost as we apply the step-forward approach. To reduce the costs, we may allow malicious agents to collude together to obtain a shared malicious update by split learning.

The fraction of malicious agents η. This parameter naturally affects the performance of Chironex. When $\eta = 20\%$ (see Table 3.3), we see that Chironex can achieve the highest ASR without any loss on MA. But if we decrease η, the ASR clearly

experiences a decline. We may keep reducing the η (i.e. increasing the number of benign agents) to $< 1\%$, then the ASR drops more seriously. This is because the impact on the backdoor task of malicious parameters is naturally diminished as the number of benign agents increases, see Table 3.3.

Defenses. We used popular and well-studied defenses, instead of all existing countermeasures, to evaluate the proposed attack's performance. The interested readers may leverage other defenses to test Chironex and the performance could not be significantly affected. We take FLTrust [38] as an example. It uses a small dataset to train a reference model and further compares local updates with the model via cosine distance. Since the distance between malicious and benign updates are still close to each other under Chironex, we can still fool the FLTrust's aggregator.

Durability. The focus of Chironex relies on stealthiness rather than persistence (e.g., Neurotoxin [19]) on attack effectiveness. Neurotoxin manipulates malicious parameters based on gradients in magnitude, which is different from Chironex focusing on constraining the contributions of backdoor neurons (by smoothing their contributions to other "less-influence" neurons). It produces a clear increase in the dissimilarity of parameters and thus it cant provide the same level of stealthiness as ours. The dissimilarity difference can be addressed in Chironex by constraining the contribution of backdoor neuron parameters (i.e, reducing the cosine dissimilarity between benign and malicious parameters). We state that persistence is orthogonal with the main focus of this work, and we leave it as an open problem. A possible solution to achieve persistence could be to decelerate the learning rate of malicious agents as in [14].

REFERENCES

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson and B. A. y Arcas. 'Communication-efficient learning of deep networks from decentralized data'. In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 1273–1282.

[2] S. Niknam, H. S. Dhillon and J. H. Reed. 'Federated learning for wireless communications: Motivation, opportunities, and challenges'. In: *IEEE Communications Magazine* 58.6 (2020), pp. 46–51.

[3] Y. Liu, Y. Kang, T. Zou, Y. Pu, Y. He, X. Ye, Y. Ouyang, Y.-Q. Zhang and Q. Yang. 'Vertical federated learning: Concepts, advances, and challenges'. In: *IEEE Transactions on Knowledge and Data Engineering* (2024).

[4] L. Fu, H. Zhang, G. Gao, M. Zhang and X. Liu. 'Client selection in federated learning: Principles, challenges, and opportunities'. In: *IEEE Internet of Things Journal* (2023).

[5] R. Myrzashova, S. H. Alsamhi, A. V. Shvetsov, A. Hawbani and X. Wei. 'Blockchain meets federated learning in healthcare: A systematic review with challenges and opportunities'. In: *IEEE Internet of Things Journal* 10.16 (2023), pp. 14418–14437.

[6] J. Zhang, S. Guo, J. Guo, D. Zeng, J. Zhou and A. Y. Zomaya. 'Towards data-independent knowledge transfer in model-heterogeneous federated learning'. In: *IEEE Transactions on Computers* 72.10 (2023), pp. 2888–2901.

[7] H. Li, Z. Cai, J. Wang, J. Tang, W. Ding, C.-T. Lin and Y. Shi. 'Fedtp: Federated learning by transformer personalization'. In: *IEEE transactions on neural networks and learning systems* (2023).

[8] M. Kesici, B. Pal and G. Yang. 'Detection of False Data Injection Attacks in Distribution Networks: A Vertical Federated Learning Approach'. In: *IEEE Transactions on Smart Grid* (2024).

[9] I. Dayan, H. R. Roth, A. Zhong, A. Harouni, A. Gentili, A. Z. Abidin, A. Liu, A. B. Costa, B. J. Wood, C.-S. Tsai *et al.* 'Federated learning for predicting clinical outcomes in patients with COVID-19'. In: *Nature medicine* 27.10 (2021), pp. 1735–1743.

[10] A. Nguyen, T. Do, M. Tran, B. X. Nguyen, C. Duong, T. Phan, E. Tjiputra and Q. D. Tran. 'Deep federated learning for autonomous driving'. In: *2022 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2022, pp. 1824–1830.

[11] G. Baruch, M. Baruch and Y. Goldberg. 'A little is enough: Circumventing defenses for distributed learning'. In: *Advances in Neural Information Processing Systems* 32 (2019).

[12] A. N. Bhagoji, S. Chakraborty, P. Mittal and S. Calo. ‘Analyzing federated learning through an adversarial lens’. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 634–643.

[13] C. Xie, K. Huang, P.-Y. Chen and B. Li. ‘Dba: Distributed backdoor attacks against federated learning’. In: *International Conference on Learning Representations*. 2019.

[14] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin and V. Shmatikov. ‘How to backdoor federated learning’. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 2938–2948.

[15] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee and D. Papailiopoulos. ‘Attack of the tails: Yes, you really can backdoor federated learning’. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 16070–16084.

[16] H. Li, Q. Ye, H. Hu, J. Li, L. Wang, C. Fang and J. Shi. ‘3dfed: Adaptive and extensible framework for covert backdoor attack in federated learning’. In: *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2023, pp. 1893–1907.

[17] H. Zhang, J. Jia, J. Chen, L. Lin and D. Wu. ‘A3fl: Adversarially adaptive backdoor attacks to federated learning’. In: *Advances in Neural Information Processing Systems* 36 (2024).

[18] R. J. Campello, D. Moulavi and J. Sander. ‘Density-based clustering based on hierarchical density estimates’. In: *Pacific-Asia conference on knowledge discovery and data mining*. Springer. 2013, pp. 160–172.

[19] Z. Zhang, A. Panda, L. Song, Y. Yang, M. Mahoney, P. Mittal, R. Kannan and J. Gonzalez. ‘Neurotoxin: Durable Backdoors in Federated Learning’. In: *Proceedings of the 39th International Conference on Machine Learning*. 2022, pp. 26429–26446.

[20] D. Yin, Y. Chen, R. Kannan and P. Bartlett. ‘Byzantine-robust distributed learning: Towards optimal statistical rates’. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 5650–5659.

[21] M. S. Ozdayi, M. Kantarcioglu and Y. R. Gel. ‘Defending against backdoors in federated learning with robust learning rate’. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 10. 2021, pp. 9268–9276.

[22] T. D. Nguyen, P. Rieger, H. Chen, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen, A. Mirhoseini, S. Zeitouni, F. Koushanfar, A.-R. Sadeghi and T. Schneider. ‘FLAME: Taming Backdoors in Federated Learning’. In: *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 1415–1432. ISBN: 978-1-939133-31-1.

[23] P. Rieger, T. D. Nguyen, M. Miettinen and A.-R. Sadeghi. ‘DeepSight: Mitigating backdoor attacks in federated learning through deep model inspection’. In: *NDSS*. 2022.

[24] D. Yin, R. Gontijo Lopes, J. Shlens, E. D. Cubuk and J. Gilmer. 'A fourier perspective on model robustness in computer vision'. In: *Advances in Neural Information Processing Systems* 32 (2019).

[25] Z.-Q. J. Xu, Y. Zhang and Y. Xiao. 'Training behavior of deep neural network in frequency domain'. In: *Neural Information Processing: 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12–15, 2019, Proceedings, Part I 26*. Springer. 2019, pp. 264–274.

[26] T. Wang, Y. Yao, F. Xu, S. An, H. Tong and T. Wang. 'An invisible black-box backdoor attack through frequency domain'. In: *European Conference on Computer Vision*. Springer. 2022, pp. 396–413.

[27] Y. Feng, B. Ma, J. Zhang, S. Zhao, Y. Xia and D. Tao. 'Fiba: Frequency-injection based backdoor attack in medical image analysis'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 20876–20885.

[28] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. McMahan, S. Patel, D. Ramage, A. Segal and K. Seth. 'Practical secure aggregation for federated learning on user-held data. arXiv 2016'. In: *arXiv preprint arXiv:1611.04482* 13 ().

[29] B. Zhao, P. Sun, T. Wang and K. Jiang. 'Fedinv: Byzantine-robust federated learning by inversing local model updates'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 8. 2022, pp. 9171–9179.

[30] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar and V. Smith. 'Federated optimization in heterogeneous networks'. In: *Proceedings of Machine learning and systems* 2 (2020), pp. 429–450.

[31] Y. Liu, Y. Kang, X. Zhang, L. Li, Y. Cheng, T. Chen, M. Hong and Q. Yang. 'A communication efficient collaborative learning framework for distributed features'. In: *arXiv preprint arXiv:1912.11187* (2019).

[32] T. Li, S. Hu, A. Beirami and V. Smith. 'Ditto: Fair and robust federated learning through personalization'. In: *International conference on machine learning*. PMLR. 2021, pp. 6357–6368.

[33] V. Tolpegin, S. Truex, M. E. Gursoy and L. Liu. 'Data poisoning attacks against federated learning systems'. In: *European Symposium on Research in Computer Security*. Springer. 2020, pp. 480–501.

[34] Y. Dai and S. Li. 'Chameleon: Adapting to peer images for planting durable backdoors in federated learning'. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 6712–6725.

[35] A. Reisizadeh, F. Farnia, R. Pedarsani and A. Jadbabaie. 'Robust federated learning: The case of affine distribution shifts'. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 21554–21565.

[36] V. Shejwalkar and A. Houmansadr. 'Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning'. In: *NDSS*. 2021.

[37] X. Cao, J. Jia and N. Z. Gong. 'Provably secure federated learning against malicious clients'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 8. 2021, pp. 6885–6893.

[38] X. Cao, M. Fang, J. Liu and N. Gong. 'FLTrust: Byzantine-robust Federated Learning via Trust Bootstrapping'. In: *Proceedings of NDSS*. 2021.

[39] C. Xie, M. Chen, P.-Y. Chen and B. Li. 'Crfl: Certifiably robust federated learning against backdoor attacks'. In: *International Conference on Machine Learning*. PMLR. 2021, pp. 11372–11382.

[40] Z. Zhang, X. Cao and N. Z. Gong. 'FLDetector: Defending Federated Learning Against Model Poisoning Attacks via Detecting Malicious Clients'. In: *KDD*. 2022.

[41] X. Fang and M. Ye. 'Robust Federated Learning With Noisy and Heterogeneous Clients'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 10072–10081.

[42] P. Blanchard, E. M. El Mhamdi, R. Guerraoui and J. Stainer. 'Machine learning with adversaries: Byzantine tolerant gradient descent'. In: *Advances in Neural Information Processing Systems* 30 (2017).

[43] R. Guerraoui, S. Rouault *et al.* 'The hidden vulnerability of distributed learning in byzantium'. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 3521–3530.

[44] Z. Sun, P. Kairouz, A. T. Suresh and H. B. McMahan. 'Can you really backdoor federated learning?' In: *arXiv preprint arXiv:1911.07963* (2019).

[45] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli and A. Anandkumar. 'signSGD: Compressed Optimisation for Non-Convex Problems'. In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by J. Dy and A. Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, 2018, pp. 560–569.

[46] A. Panda, S. Mahloujifar, A. N. Bhagoji, S. Chakraborty and P. Mittal. 'Sparsefed: Mitigating model poisoning attacks in federated learning with sparsification'. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2022, pp. 7587–7624.

[47] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru and B. Li. 'Manipulating machine learning: Poisoning attacks and countermeasures for regression learning'. In: *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2018, pp. 19–35.

[48] Y. LeCun. 'The MNIST database of handwritten digits'. In: *http://yann. lecun. com/exdb/mnist/* (1998).

[49] H. Xiao, K. Rasul and R. Vollgraf. 'Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms'. In: *arXiv preprint arXiv:1708.07747* (2017).

[50] A. Krizhevsky, G. Hinton *et al.* 'Learning multiple layers of features from tiny images'. In: (2009).

[51] S. Caldas, S. M. K. Duddu, P. Wu, T. Li, J. Konen, H. B. McMahan, V. Smith and A. Talwalkar. 'Leaf: A benchmark for federated settings'. In: *arXiv preprint arXiv:1812.01097* (2018).

[52] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei. 'ImageNet Large Scale Visual Recognition Challenge'. In: *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252. DOI: 10.1007/s11263-015-0816-y.

[53] 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.

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

[55] V. Shejwalkar, A. Houmansadr, P. Kairouz and D. Ramage. 'Back to the Drawing Board: A Critical Evaluation of Poisoning Attacks on Production Federated Learning'. In: *2022 IEEE Symposium on Security and Privacy (SP)* (2021), pp. 1354–1371.

[56] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei. 'ImageNet: A large-scale hierarchical image database'. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[57] K. Simonyan and A. Zisserman. 'Very Deep Convolutional Networks for Large-Scale Image Recognition'. In: *International Conference on Learning Representations*. 2015.

4

GENERATIVE BACKDOOR ATTACK AGAINST FEDERATED LEARNING

*Current backdoor attacks against federated learning (FL) manipulate inputs with universal triggers or semantic patterns, which can be detected and filtered by certain defense mechanisms such as norm clipping, trigger inversion. In this work, we propose a generator-assisted backdoor attack with flexible triggers, FTA, against FL defenses. We consider the stealthiness of both input and feature space of triggers under decentralized settings. In this method, we propose a generative trigger function that can learn to manipulate the benign samples with naturally imperceptible trigger patterns and simultaneously minimize representation distance between poisoned and benign samples under the attacker-chosen label (*stealthiness*). Moreover, our trigger generator repeatedly produces triggers for each sample (*flexibility*) in each FL iteration (*adaptivity*), allowing it to adjust to changes of feature representations between global models of different rounds. Instead of using universal triggers of existing works, we break this wall by providing three desiderata (i.e., stealthiness, flexibility and adaptivity), which helps our attack avoid the presence of backdoor-related feature representations. Extensive experiments confirm the effectiveness (above 98% attack success rate) and superior stealthiness of our attack compared to prior attacks under seven well-studied FL defenses.*

This chapter is based on the paper "FTA: Stealthy Backdoor Attack with Flexible Triggers against Federated Learning." by Qiao, Y., Liu, D., Panaousis, M., Conti, M., and Liang, K., which is under review from IEEE Transactions on Artificial Intelligence (2025).

4.1. INTRODUCTION

Federated learning (FL) has recently provided practical performance in various real-world applications and tasks, such as prediction of oxygen requirements of symptomatic patients with COVID-19 [1], autonomous driving [2], Gboard [3] and Siri [4]. It supports collaborative training of an accurate global model by allowing multiple agents to upload local updates, such as gradients or weights, to a server without compromising local datasets. However, this decentralized paradigm unfortunately exposes FL to a security threat backdoor attacks [5–9]. Existing backdoor defenses on FL possess the capability to scrutinize the anomaly of malicious model updates. Prior attacks fail to achieve adequate stealthiness under those robust FL systems due to malicious parameters tuning introduced by the backdoor task.

We summarize the following open problems from the existing backdoor attacks against FL:

P1: The abnormality of feature extraction in convolutional layers. Existing attacks use patch-based triggers (“squares", “stripe" and etc.) [6, 8–10] on a fixed position or semantic backdoor triggers (shared attributes within the same class) [7, 10]. However, we found that these triggers fail to provide enough “stealthiness" of the hidden features of the poisoned samples. This is so because latent representations of poisoned samples extracted from filters *standalone* compared to the benign counterparts. **Figure 4.10 (a)** intuitively illustrates the statement. This abnormality induces weight outliers in the parameter space.

P2: The abnormality of backdoor routing in fully connected layers. In fully-connected (FC) layers, the backdoor task is to establish a *new* routing [11, 12], separated from benign ones, between the independent hidden features of attacker's trigger pattern and its corresponding target label, which yields an anomaly at the parameter level. The cause of this anomaly is natural, since the output neurons for the target label must contribute to both benign and backdoor routing, which requires significant weight/bias adjustments to the neurons involved. Therefore, backdoor routing can be seen as the secondary source of these abnormalities. Note that these abnormalities (**P1-2**) would arise in existing universal trigger designs under FL.

P3: The perceptible trigger for inference. Perhaps, it is not necessary to guarantee natural stealthiness of triggers on training data against FL, since its accessibility is limited to each client exclusively due to the privacy issue. However, the test input with perceptible perturbation in FL [6, 8–10] can be easily identified by an evaluator or a user who can distinguish the difference between ‘just an incorrect classification/prediction of the model and the purposeful wrong decision due to a backdoor in the test/use stage.

In this work, we regard the problems **P1-3** as the *stealthiness* of backdoor attacks in the context of FL. A natural question then arises: *could we eliminate the anomalies introduced by new backdoor features and routing* ***(i.e., tackling P1-2)*** *while making the trigger sufficiently stealthy for inference on decentralized scenario* ***(i.e., addressing P3)****?*

To provide a concrete answer, we propose a stealthy generator-assisted backdoor attack, FTA, to adaptively (per FL iteration) provide triggers in a flexible manner

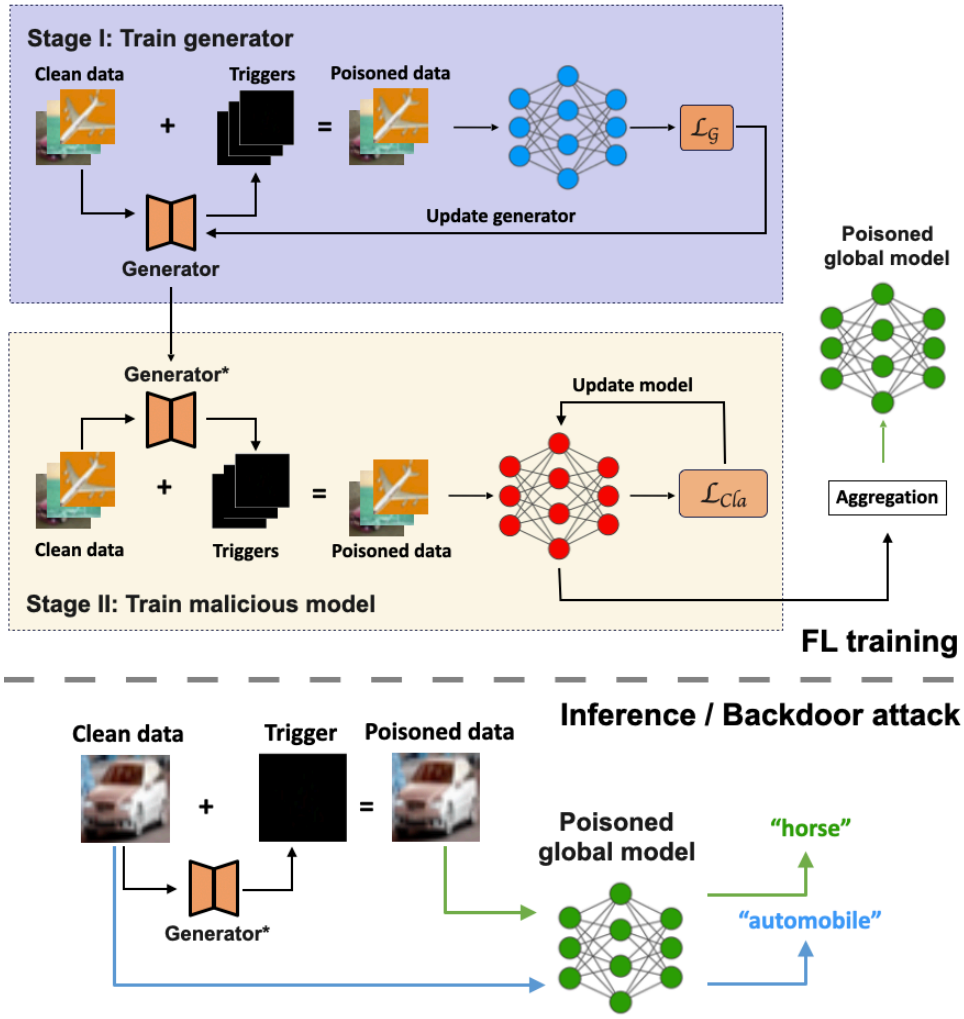


Figure 4.1: Overview of FTA. (I) Learn the optimal trigger generator g_ξ. (II) Train malicious model f_θ. Inference/Backdoor Attack: The global model performs well on benign tasks while misclassifying the poisoned samples to the target label.

(per sample) on decentralized setup. FTA achieves a satisfied stealthiness by producing imperceptible triggers with a generative neural network (GAN) [13, 14] in a *flexible* way for each sample and in an *adaptive* manner during entire FL iterations. To address **P3**, our triggers should provide natural stealthiness to avoid inspection during inference. To solve **P1**, the difference of hidden representation between poisoned data and benign counterparts should be minimized. Due to the imperceptibility between poisoned and benign data in latent representation,

the correspondent backdoor routing will not be formed and thus **P2** is effectively addressed.

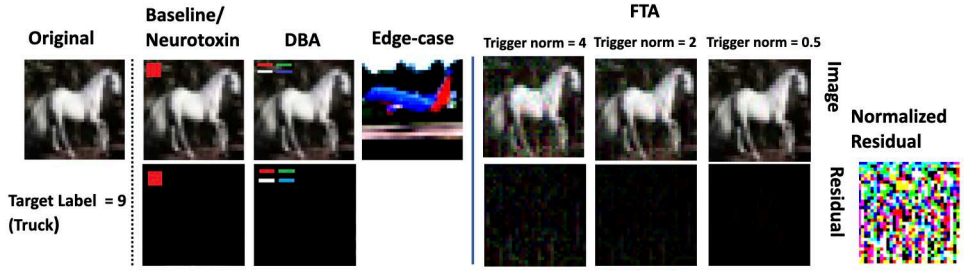


Figure 4.2: Visualization of backdoored images. **Top**: the original image; backdoored samples generated by baseline/Neurotoxin, DBA, Edge-case, and FTA; **Bottom**: the residual maps. Our flexible triggers appear as imperceptible noise.

Specifically, the proposed generator is learned to minimize the representation distance for each poisoned sample between output prediction and the target label, which can ensure similar feature representations of poisoned samples to benign ones with target label **(P1)**. This can also reduce the abnormality of creating an extra routing for backdoor in **P2** since the similar features make poisoned data “look like" benign ones with target label, allowing poisoned data to reuse the benign routing. As a result, the backdoor task does not need to be learned entirely from scratch, thereby achieving high attack efficiency, as shown in Figure 4.3. Meanwhile, our trigger is less perceptible and more flexible than predefined patch-based ones in prior attacks **(P3)**. Further, to make the flexible trigger robust and adaptive to the changes in global model, the generator is continuously trained across FL iterations. Finally, we formulate the process of finding optimal trigger generator and training malicious model in a bi-level and constrained optimization problem, and achieve the optimum by proposing a simple but efficient optimization process. Compared with existing works using patch-wise triggers, we break this wall and for the first time make the generated trigger stealthy, flexible and adaptive in FL setups. We illustrate learning the trigger generator, training the malicious model and testing the backdoor in Figure 4.1, and showcase various backdoor images in Figure 4.2 to demonstrate the imperceptible perturbation by our generator.

Our main **contributions** are summarized as follows:

- We propose a stealthy generator-assisted backdoor attack (FTA) against robust FL. Instead of utilizing an universal trigger pattern, we design a novel trigger generator that produces naturally imperceptible triggers during inference stage. Our flexible triggers provide hidden feature similarity of benign data and successfully lead poisoned data to reuse benign routing of target label. Hereby FTA can avoid anomaly in parameter space and improve attack effectiveness.
- We design a new learnable and adaptive generator that can learn the flexible triggers for global model at current FL iteration to achieve the best attack

effectiveness. We propose a bi-level and constrained optimization problem to find our optimal generator each iteration efficiently. We then formulate a customized learning process and solve it with reasonable complexity, making it applicable to the FL scenario.

• Finally, we present intensive experiments to empirically demonstrate that the proposed attack provides state-of-the-art effectiveness and stealthiness against eight well-study defense mechanisms under four benchmark datasets.

4.2. RELATED WORK

4.2.1. FEDERATED LEARNING

Consider the empirical risk minimization (ERM) in FL setting where the goal is to learn a global classifier $f_\theta : \mathcal{X} \to \mathcal{Y}$ that maps an input $x \in \mathcal{X}$ to a target label $y \in \mathcal{Y}$. Recall that the FL server cannot access the training dataset. It aggregates the parameters/gradients from local agents performing centralized training with local datasets. The de-facto standard rule for aggregating the updates is so-called `FedAvg` [15]. The training task is to learn the global parameters θ by solving the finite-sum optimization: $\min_\theta \ f_\theta = \frac{1}{n}\sum_{i=1}^{n} f_{\theta_i}$, where n is the number of participating agents. At round t, the server S randomly selects $n^t \in \{1,2,...,n\}$ agents to participate in the aggregation and send the global model θ^t to them. Each of the agents i trains its local classifier $f_{\theta_i} : \mathcal{X}_i \to \mathcal{Y}_i$ with its local dataset $D_i = \{(x_j, y_j) : x_j \in \mathcal{X}_i, y_j \in \mathcal{Y}_i, j = 1,2,...,N\}$ for some epochs, where $N = |D_i|$, by certain optimization algorithm, e.g., stochastic gradient descent (SGD). The objective of agent i is to train a local model as: $\theta_i^* = \underset{\theta^t}{\operatorname{argmin}} \sum_{(x_j,y_j)\in D_i} \mathcal{L}(f_{\theta^t}(x_j), y_j)$, where $\mathcal{L}$ stands for the classification loss, e.g., cross-entropy loss. Then agent i computes its local update as $\delta_i^t = \theta_i^* - \theta^t$, and sends back to S. Finally, the server aggregates all updates and produces the new global model with an average $\theta^{t+1} = \theta^t + \frac{\gamma}{|n^t|}\sum_{i\in n^t} \delta_i^t$. where γ is the global learning rate. When the global model θ converges or the training reaches a specific iteration upper bound, the aggregation process terminates and outputs a final global model. During inference, given a benign sample x and its true label y, the learned global classifier f_θ will behave well as: $f_\theta(x) = y$.

Optimizations of FL have been proposed for various purposes, e.g., privacy [16], security [17, 18], heterogeneity [19], communication efficiency [20, 21] and personalization issues [22, 23].

4.2.2. BACKDOOR ATTACKS ON FL

The most well-known backdoor attack on FL is introduced in [10], where the adversary scales up the weights of malicious model updates to maximize attack impact and replace the global model with its malicious local model. To fully exploit the distributed learning methodology of FL, the local trigger patterns are used in [6] to generate poisoned images for different malicious models, while the data from the tail of the input data distribution is leveraged in [7]. Durable backdoor attacks are proposed in [8], and make attack itself more persistent in the federated scenarios.

We state that this kind of attacks mainly focuses on the persistence, whereas our focus is on stealthiness.

Existing works rely on a universal trigger or tail data, which do not fully exploit the “attribute" of trigger. Our design is fully applicable and complementary to prior attacks. By learning a stealthy trigger generator and injecting the sample-specific triggers, we can significantly decrease the anomalies in **P1-3** and reinforce the stealthiness of backdoor attacks.

4.2.3. BACKDOOR DEFENSES ON FL

There are a number of defenses that provide empirical robustness against backdoor attacks.

Dimension-wise filtering. Trimmed-mean [24] aggregates each dimension of model updates of all agents independently. It sorts the parameters of the j^{th}-dimension of all updates and removes m of the largest and smallest parameters in that dimension. Finally, it computes the arithmetic mean of the rest parameters as the aggregate of dimension j. Similarly, Median [24] takes the arithmetic median value of each dimension for aggregation. SignSGD [25] only aggregates the signs of the gradients (of all agents) and returns the sign to agents for updating the local models.

Vector-wise scaling. Norm clipping [26] bounds the l_2-norm of all updates to a fixed threshold due to high norms of malicious updates. For a threshold τ and an update ∇, if the norm of the update $||\nabla|| > \tau$, ∇ is scaled by $\frac{\tau}{||\nabla||}$. The server averages all the updates, scaled or not, for aggregation.

Vector-wise filtering. Krum [17] selects a local model, with the smallest Euclidean distance to $n-f-1$ of other local models, as the global model. A variant of Krum called Multi-Krum [17] selects a local model using Krum and removes it from the remaining models repeatedly. The selected model is added to a selection S until S has c models such that $n-c > 2m+2$, where n is the number of selected models and m is the number of malicious models. Finally, Multi-Krum averages the selected model updates. RFA [27] aggregates model updates and makes FedAvg robust to outliers by replacing the averaging aggregation with an approximate geometric median.

Certification. CRFL [28] provides certified robustness in FL frameworks. It exploits parameter clipping and perturbing during federated averaging aggregation. In the test stage, it constructs a “smoothed" classifier using parameter smoothing. The robust accuracy of each test sample can be certified by this classifier when the number of compromised clients or perturbation to the test input is below a certified threshold.

Sparsification. SparseFed [29] performs norm clipping to all local updates and averages the updates as the aggregate. Top_k values of the aggregation update are extracted and returned to each agent who locally updates the models using this sparse update.

Cluster-based filtering. Recently, [30] proposed a defending framework FLAME based on the clustering algorithm (HDBSCAN) which can cluster dynamically all local updates based on their cosine distance into two groups separately. FLAME uses weight clipping for scaling-up malicious weights and noise addition for smoothing

the boundary of clustering after filtering malicious updates. By using HDBSCAN, [31] designed a robust FL aggregation rule called DeepSight. Their design leverages the distribution of labels for the output layer, output of random inputs, and cosine similarity of updates to cluster all agents' updates and further applies the clipping method.

4.3. THREAT MODEL AND INTUITION

4.3.1. THREAT MODEL

Attacker's Knowledge & Capabilities: We consider the same threat model as in prior works [5, 7, 8, 10, 29, 32], where the attacker can have full access to malicious agent device(s), local training processes and training datasets. Furthermore, we do not require the attacker to know the FL aggregation rules applied in the server.

Attacker's Goal: Unlike untargeted poisoning attacks [33] preventing the convergence of the global model, the goal of our attack is to manipulate malicious agents' local training processes to achieve high accuracy in the backdoor task without undermining benign accuracy.

4.3.2. OUR INTUITION

Recall that prior attacks use universal predefined patterns (see Figure 4.2) which cannot guarantee stealthiness (**P1-3**) since the poisoned samples are visually inconsistent with natural inputs. These triggers (including tail data) used in whole FL iterations with noticeable modification can introduce new hidden features during extraction and further influence the process of backdoor routing. Consequently, this makes prior attacks be easily detected by current robust defenses due to **P1-2**. Also, the inconsistency between benign and poisoned samples is not stealthy for the attacker during the global model inference (**P3**) and the triggers can be inversed in decentralized setup.

Compared to prior attacks that focus on manipulating parameters, we bridge the gap and focus on designing stealty triggers. To address **P1-3**, a well-designed trigger should provide 4 superiorities: 1) the poisoned sample is naturally stealthy to the original benign sample; 2) the trigger is able to achieve feature similarity between poisoned and benign samples of target label; 3) the trigger can eliminate the anomaly between backdoor and benign routing during learning; 4) the trigger design framework can evade robust FL defenses. The only solution that provides these advantages over prior works simultaneously is *flexible* triggers. The optimal flexible triggers are learnt to make latent representations of poisoned samples similar to benign ones and thus make the reuse of benign routing possible, which naturally diminish the presence of outlier at parameter level. Therefore, to achieve the flexibilty of trigger patterns and satisfy four requirements, we propose a learnable and adaptive trigger generator to produce flexible and stealthy triggers.

v.s. Trigger generators in centralized setting. One may argue that the attacker can simply apply a similar (trigger) generator in centralized setup [34–38] on FL to achieve imperceptible trigger and stealthy model update.

• **Stealthiness.** For example, the attacker can use a generator to produce imperceptible triggers for poisoned samples and make their hidden features similar to original benign samples' as in [36, 38]. This, however, cannot ensure the indistinguishable perturbation of model parameters (caused by backdoor routing) during malicious training and fail to capture the stealthiness (in **P1-2**). This is so because it only constrains the distinction of the input domain and the hidden features between poisoned and benign samples other than the hidden features between poisoned and benign samples of *target* label. In other words, a centralized generator masks triggers in the input domain and feature space of benign samples, conceals the poisoned sample for visibility and feature representation, whereas this cannot ensure the absence of backdoor routing for poisoned data. A stealthy backdoor attack on FL should mitigate the routing introduced by backdoor task training and guarantee the stealthiness of model parameters instead of just the hidden features of poisoned samples compared to their original inputs.

• **Learning.** The centralized learning process of existing trigger generator cannot directly apply to decentralized setups due to the continuously changing of global model and time consumption of training trigger generator. As an example, IBA [38] directly constrains the distance of feature representation between benign and poisoned samples. This approach cannot achieve satisfied attack effectiveness due to the inaccurate hidden features of benign samples before global model convergence. In contrast, we propose a customized optimization method for the FL scenario that can learn the optimal trigger generator for global model of current iteration to achieve the best attack effectiveness and practical computational cost as depicted in Section 4.4.3 and Table 4.3, respectively.

• **Defenses.** We note that the robust FL aggregator can only access local updates of all agents other than local training datasets. The centralized backdoor attack does not require consideration of the magnitude of the malicious parameters. However, in reality, the magnitude of malicious updates is usually larger than that of benign updates under FL setups. In that regard, norm clipping can effectively weaken and even eliminate the impact of the backdoor [26, 32]. Thanks to the flexibility of our triggers, we advance the state-of-the-art by enhancing the stealthiness and effectiveness of the backdoor attack even against well-studied defenses such as trigger inversion method on FL, e.g. FLIP [39]. FLIP is effective in removing prior backdoors with patch-based triggers whereas our attack can naturally evade this SOTA FL defense.

4.4. PROPOSED METHODOLOGY: FTA

4.4.1. FTA TRIGGER FUNCTION

We first introduce our trigger function $\mathcal{T} : \mathcal{X} \to \mathcal{X}$ with a given generative network, i.e., our trigger generator. To eliminate three abnormalities of **P1-3**, the key insight of our trigger function is to guarantee: 1) The perturbations of poisoned sample are imperceptible; and 2) The trigger generator can effectively learn representations of the target label for flexible triggers. Given a benign sample x and corresponding

label y, we formally model $\mathcal{T}_\xi$ and target labeling function η as follows:

$$x' = \mathcal{T}_\xi(x) = x + g_\xi(x), \quad y' = \eta(y) = c, \tag{4.1}$$

where g is the generator, ξ is the parameters of g, and c is the specific target label. We use the same neural network architectures as [34] to build our trigger generator, i.e., autoencoder and U-Net architectures [40].

4.4.2. PROBLEM FORMULATION

Based on the federated scenario in Section 4.2.1, the attacker's main objective is to train the malicious models in order to alter the behavior of the global model f (with parameters θ) as follows:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{(x,y)\in D_{cln}} \mathcal{L}(f_\theta(x), y) + \sum_{(x',y')\in D_{bd}} \mathcal{L}(f_\theta(x'), y'), \tag{4.2}$$

where D_{cln} is clean local dataset and D_{bd} is a small fraction of clean samples in D_{cln} to produce poisoned data via Equation (4.1). And the poison ratio is defined as $|D_{bd}|/|D_{cln}|$. During inference, given a clean input x and its true label y, the victim model f behaves as: $f(x) = y, f(\mathcal{T}(x)) = \eta(y)$.

To eliminate the anomaly introduced by backdoor representations, our additional goal is to learn a stealthy trigger generator g to produce flexible and stealthy triggers with the following: 1) the generated triggers ensure that poisoned images do not bring visual disparities compared to benign ones; 2) the poisoned global model simultaneously performs indifferently on test input x but changes its prediction on the poisoned sample x' to the target label; and 3) the feature representation of backdoor sample $\mathcal{T}(x)$ is similar to its benign input x. To achieve similar feature representations under target label, we here introduce representation distance between the model prediction of input and the target class as follows:

$$\mathcal{R}(x, y, f_\theta) = \| f_\theta(x)_L - O(y) \|_2, \tag{4.3}$$

where $f_\theta(x)_L$ is the logits of the output layer L, $O(y)$ is the one-hot vector of label y. It approximates the similarity between input sample and the target label in the representation space. By minimizing $\mathcal{R}(x', c, f_\theta)$, we can force the generator to learn the feature representations of target label from victim model into trigger pattern for each poisoned sample.

Therefore, the overall attack objectives of FTA can be formulated as a constrained optimization as follows:

$$\begin{aligned} &\min_{\theta,\xi} \sum_{(x,y)\in D_{cln}} \mathcal{L}(f_{\theta^t}(x), y) \\ &+ \sum_{(x',y')\in D_{bd}} \mathcal{L}(f_{\theta^t}(x'), y') \\ s.t. \quad &(i) \quad \xi^* = \underset{\xi}{\operatorname{argmin}} \sum_{(x',y')\in D_{bd}} \mathcal{R}(x', y', f_{\theta^t})) \\ &(ii) \quad d(x', x) \le \epsilon \end{aligned}, \tag{4.4}$$

where t is current FL round, d is l_2-norm distance, ϵ is a constant threshold value to ensure a minimal perturbation. In the above bilevel problem, we optimize malicious model f_{θ^t} that is associated with a generative trigger function $\mathcal{T}_\xi$. However, solving this problem in Equation (4.4) is not trivial under FL scenario since the global model f_θ varies in each iteration and the non-linear constraint. Thus, the learned trigger function $\mathcal{T}_\xi$ is unstable based on dynamic f_θ. The detailed optimization procedure is depicted in the following subsection.

4.4.3. FTA'S OPTIMIZATION

To address the constrained optimization in Equation (4.4), existing function-based attacks [34, 36, 41] alternately updating f_θ while keeping g_ξ unchanged, or the other way round, for many iterations. However, it involves a large amount of computational cost and makes the training process unstable according to our trials. Thus, we divide FTA optimization into two phases within a single FL round, with each phase executed in only one iteration over a few epochs. In phase one, we fix the global model f_{θ^t} and only learn the trigger generator g_ξ. In phase two, we use the pretrained g_ξ to generate D_{bd} and train the malicious classifier f_{θ^t}. Note that we reuse generator g_ξ trained in t-th round for the next attack round. By doing so, we prevent the generator g_ξ from overfitting, enabling it to be continuously learned to extract feature representations of the target label from dynamic global models across different rounds. Furthermore, this mechanism allows the malicious model to reuse the benign routing for poisoned samples without significantly tuning the parameters for backdoor task (achieving **P1-2**). Algorithm 5 describes the details of optimization procedure of FTA in t-th FL round.

4.5. ATTACK EVALUATION

We show that FTA outperforms 3 SOTA attacks (under 8 robust FL defenses) by conducting experiments on different computer vision tasks.

4.5.1. EXPERIMENTAL SETUP

Datasets and Models. We demonstrate the effectiveness of FTA backdoor through comprehensive experiments on four publicly available datasets, namely Fashion-MNIST (FMNIST) [42], FEMNIST [43], CIFAR-10 [44], and Tiny-ImageNet (TI) [45]. In Fashion-MNIST, CIFAR-10 and Tiny-ImageNet, a Dirichlet distribution with a degree of 0.7 is used to divide training data for the number of total agent parties. For FEMNIST, we randomly choose data of 3000 users from the dataset and randomly distribute every training agent with the training data from 3 users. The classification model used in the experiments includes Classic CNN, VGG11 [46], and ResNet18 [47]. These datasets and models are representative and commonly used in existing backdoor and FL research works.

Algorithm 5: FTA Optimization

Input: Clean dataset D_{cln}, Global model f with parameters θ^t, Trigger generator g with parameters ξ, Learning rates γ_f and γ_g, Batch of clean and poisoned dataset B_{cln} and B_{bd}, Number of Epochs e_f and e_g
Output: Malicious model update δ^m
Initialize parameters θ^t and ξ
$D_{bd} \leftarrow$ Poison a random subset of D_{cln} by Equation (4.1)
// Stage I: Update flexible generator g
Sample minibatch $(x', y') \in B_{bd}$ from D_{bd}
for $i = 1, 2, \cdots, e_g$ **do**
    Optimize ξ by using SGD with fixed f_{θ^t} on B_{bd}: $\xi \leftarrow \xi - \gamma_g \nabla_\xi \mathscr{R}(x', y', f_{\theta^t})$
end
$\xi^* \leftarrow \xi$ // save for next attack round
// Stage II: Train malicious model f_{θ^t}
Sample minibatch $(x, y) \in B_{cln}$ from D_{cln} and $(x', y') \in B_{bd}$ from D_{bd}
for $i = 1, 2, \cdots, e_f$ **do**
    Optimize θ^t by using SGD with fixed g_{ξ^*} on B_{cln} and B_{bd}:
    $\theta^{t+1} \leftarrow \theta^t - \gamma_f \nabla_{\theta^t}(\mathscr{L}(f_{\theta^t}(x, y)) + \mathscr{L}(f_{\theta^t}(x', y')))$
end
Compute malicious update: $\delta^m \leftarrow \theta^{t+1} - \theta^t$
return δ^m

4.5.2. DETAILS OF THE TASKS

The details of 4 computer vision tasks are described in Table 4.1. To prove the stealthiness against defenses of FTA, we use a decentralized setting with non-i.i.d. data distribution among all agents. The attacker chooses the all-to-one type of backdoor attack (except Edge-case [7]), fooling the global model to misclassify the poisoned images of any label to an attacker-chosen target label. We apply backdoor attacks from different phases of training. In FEMNIST task, we follow the same setting as [6], where the attacker begins to attack when the benign accuracy of global models starts to converge. For other tasks, we perform backdoor attacks at the beginning of FL training. In this sense, as mentioned in [6], benign updates are more likely to share common patterns of gradients and have a larger magnitude than malicious updates, which can significantly restrict the effectiveness of malicious updates. Note we consider such a setting for the bottom performance of attacks and further, we still see that our attack performs more effectively than prior works in this case (see Figure 4.3).

Implementation. The implementation of all the attacks and FL frameworks is based on PyTorch [48]. We test all experiments on a server with one Intel Xeon E5-2620 CPU and one NVIDIA A40 GPU with 32G RAM. As in Neurotoxin [8], we assume that the attacker can only compromise a limited number of agents (<1%) in practice [32] and uses them to launch the attack by uploading manipulated gradients to the server. Following a practical scenario for the attacker given in [8], *10* agents among

Table 4.1: The datasets, and their corresponding models and hyperparameters.

	Fahion-MNIST	FEMNIST	CIFAR-10	Tiny-ImageNet
Classes	62	10	10	200
Size of training set	60000	737837	50000	100000
Size of testing set	10000	80014	10000	10000
Total agents	2000	3000	1000	2000
Malicious agents	1	2	1	1
Agents per FL round	10	10	10	10
Phase to start attack	Attack from scratch	Attack after convergence	Attack from scratch	Attack from scratch
Poison fraction	0.1			
Trigger size	1	1.5	1.5	3
Dataset size of trigger generator	1024			
Epochs of benign task	2	4	5	5
Epochs of backdoor task	5 (FTA: 2)	10 (FTA: 4)	10 (FTA: 5)	10 (FTA: 5)
Learning rate of trigger generator	0.01	0.01	0.001	0.01
Epochs of trigger generator	5	5	10	10
Local data distribution	non-i.i.d.			
Classification model	Classic CNN	Classic CNN	ResNet-18	ResNet-18
Trigger generator model	Autoencoder	Autoencoder	U-Net	Autoencoder
Learning rate of benign task	0.1	0.01	0.01	0.001
Learning rate of backdoor task	0.1	0.01	0.01	0.01
Edge-case	FALSE	TRUE	TRUE	FALSE
Other hyperparameters	Momentum:0.9, Weight Decay: 10^{-4}			

thousands of agents are selected for training in each round and their updates are used for aggregation and updating the server model. The target labels are “sneaker" in Fashion-MNIST, “digit 1" in FEMNIST, “truck" in CIFAR-10 and “tree frog" in Tiny-ImageNet. We set the poison ratio to 10%. All local clients use SGD as an optimizer and train for local training epochs with a batch size of 256. The attacker has its own local malicious learning rate and epochs to maximize its backdoor performance. It also needs to train its local trigger generator with learning rate and epochs before performing local malicious training on the downloaded global model.

Attack modes. We test stealthiness and durability of FTA with two attack modes respectively, i.e., fixed-frequency and few-shot as Neurotoxin. *(i) Fixed-frequency mode:* The server randomly chooses 10 agents among all agents. The attacker controls exactly one agent in each round in which they participate. For other rounds, 10 benign agents are randomly chosen among all agents. *(ii) Few-shot mode:* The attacker participates only in `Attack_num` rounds. During these rounds, we ensure that one malicious agent is selected for training. After Attack_num rounds or backdoor accuracy has reached 95%, the attack will stop. Under this setting, the attack can take effect quickly, and gradually weaken by benign updates after the attack is stopped.

Evaluation Metrics. We evaluate the performance based on attack success rate (ASR) and benign accuracy according to the following criteria: effectiveness and stealthiness against current SOTA defense methods under fixed-frequency mode, durability evaluated under few-shot mode.

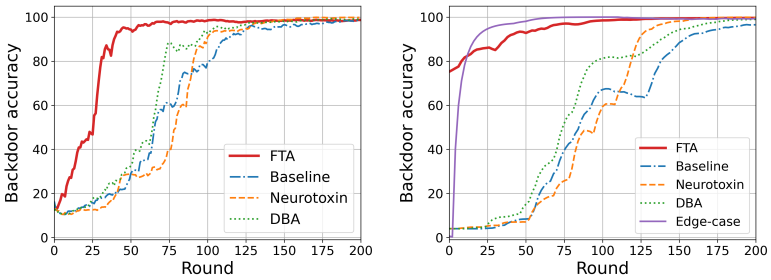
Comparison. We compare FTA with three SOTA attacks, namely DBA, Neurotoxin

and Edge-case [7], and the baseline attack method described in [8] under different settings and eight defenses (a variant of norm clipping based on [26], FLAME [30], Multi-Krum [17], Trimmed-mean [24], RFA [27], SparseFed [29], SignSGD [25] and Foolsgold [49]). Regarding the attack methods, we set the top-k ratio of 0.95 for Neurotoxin, in line with the recommended settings in [8]. For DBA, we use 4 distributed strips as backdoor trigger patterns. Both the baseline attack and Neurotoxin employ a “square" trigger pattern on the top left as the backdoor trigger. We conduct Edge-case attack on CIFAR-10 and FEMNIST. Specifically, for CIFAR-10, we use the southwest airplane as the backdoored images and set the target label as “bird". For FEMNIST, we use images of “7 in ARDIS [50] as poisoned samples with the target label set as the digit “1. The dataset settings of the experiments are the same as those used in [7]. The results demonstrate that FTA delivers the best performance as compared to others.

4.5.3. ATTACK EFFECTIVENESS

Attack effectiveness under fixed-frequency mode. Compared to the attacks with unified triggers, FTA converges much faster and delivers the best BA in all cases since our poisoned data can reuse benign routing of target label, see Figure 4.3. It can yield a high backdoor accuracy on the server model within very few rounds (<50) and maintain above 97% accuracy on average. Especially in Tiny-ImageNet, FTA reaches 100% accuracy extremely fast, with at least 25% advantage compared to others. In CIFAR-10, FTA achieves nearly 83% BA after 50 rounds which is 60% higher than other attacks on average. There is only <5% BA gap between FTA and Edge-case on FEMNIST in the beginning and later, they reach the same BA after 100 rounds. We note that the backdoor task of Edge-case in FEMNIST is relatively easy, mapping 7-like images to the target label of digit “1", which makes its convergence slightly faster than ours.

Attack effectiveness under few-shot mode. As an independent interest, we test the durability of the attacks during training stage in this setting. In our experiments, the Attack_num is 100 for all attacks, and the total FL round is 1000 for CIFAR-10, and 500 for other datasets. The results under few-shot settings are shown in Figure 4.4. All attacks reach a high BA rapidly after consistently poisoning the server model, then BA gradually drops after stopping attacking and the backdoor injected into the server model is gradually weakened by the aggregation of benign updates. FTA’s performance drops much slower than the baseline attack. For example, in Fashion-MNIST and after 500 rounds, FTA still remains 73% BA, which is only 9% less than Neurotoxin, 61% higher than the baseline. Moreover, FTA can beat DBA and the baseline on Tiny-ImageNet. After 500 rounds, FTA maintains 37% accuracy while the baseline and DBA only have 5%, which is 45% less than Neurotoxin. However, Neurotoxin cannot provide the same stealthiness as shown in following comparison under robust FL defenses. Since malicious and benign updates have a similar direction by FTA, the effectiveness of FTA’s backdoor can survive after few-shot attack. *In conclusion, FTA has long-term attack effectiveness even if we stop attacking early since the poisoned data with our well-learned triggers contain similar features to benign data and can be naturally misclassified into the target label with*



(a) Fashion-MNIST (b) FEMNIST

(c) CIFAR-10 (d) Tiny-ImageNet

Figure 4.3: Fixed-frequency attack performance under `FedAvg`. FTA is more efficient than others.

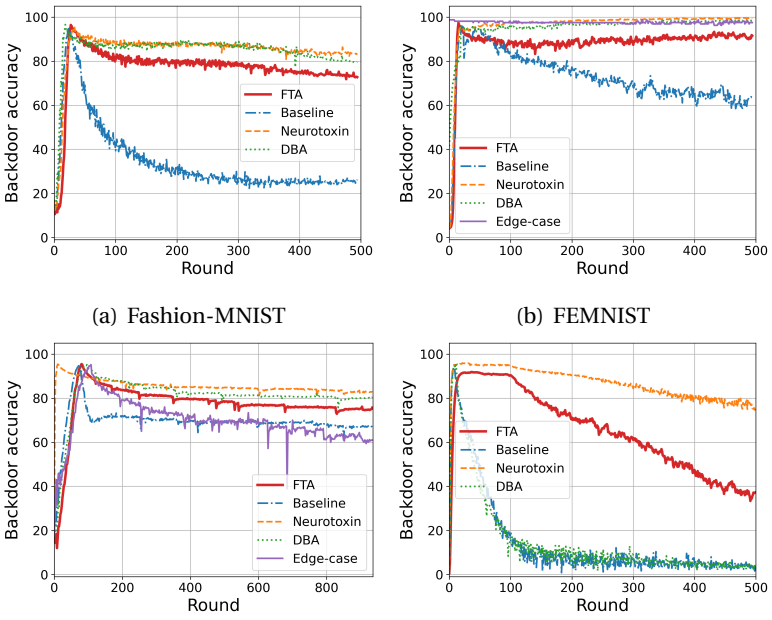
certain confidence by server model.

Influence on Benign accuracy. We showcase the benign accuracy of both the baseline attack and FTA, and also consider the accuracy without backdoor attacks under `FedAvg`. We start FTA and the baseline from a specific round (e.g., 0 or 200 for different datasets) and perform the attacks during Attack_num rounds. We record the accuracy once the attacks have ended. From Table 4.2, it is evident that FTA results in a slightly smaller decrease in benign accuracy compared to baseline attack.

Table 4.2: Benign accuracy of the baseline attack. FTA and no attackers circumstance under different datasets. Benign accuracy drops by ≤ 1.5% in FTA compared to the accuracy without attack.

Dataset	Attack start epoch	Attack_num	No attack (%)	Baseline attack (%)	FTA (%)
Fashion-MNIST	0	50	90.21	85.14	90.02
FEMNIST	200	50	92.06	91.27	92.05
CIFAR-10	0	100	61.73	56.34	60.61
Tiny-ImageNet	0	100	25.21	19.06	25.13

Computational cost. We understand the significance of the external computational cost and time consumption of backdoor training on malicious devices in our



(a) Fashion-MNIST (b) FEMNIST

(c) CIFAR-10 (d) Tiny-ImageNet

Figure 4.4: Few-shot attack performance under `FedAvg`. FTA is more durable than baseline.

proposed attack under FL scenario. Training with GANs in federated systems introduces extra time consumption. However, our attack does not significantly increase the computational and time cost due to our optimization procedure. Compared to training benign task and baseline backdoor task, FTA only needs to train an additional trigger generator which is actually a small generative neural network. Our generator only consists of several convolutional layers in total. It is worth noting that the datasets used to train both two network structures comprise only 1024 poisoned samples as shown in Table 4.1 whose sizes are relatively small compared to the entire training dataset. For instance, the training dataset for our trigger generator accounts for approximately 0.14% of the FEMNIST dataset Therefore, the time consumption and computational cost for training this generative network are very minimal. The remaining time consumption is comparable to training a benign local model. As shown in Table 4.3, Neurotoxin requires approximately 2× the time and memory compared to benign training to complete backdoor training for one FL round. This is attributed to an additional local benign training requirement in Neurotoxin. However, FTA consumes less than 30% additional time and 25% additional computational cost for backdoor training compared to benign ones. Given that FTA remains under 70% of the cost of Neurotoxin, it is practical to conduct an FTA attack in decentralized scenario.

Table 4.3: Time consumption and computational cost (MEAN±SD) of different attack methods in one FL iteration under Fashion-MNIST, CIFAR-10 and Tiny-ImageNet.

Dataset→	Fashion-MNIST		CIFAR-10		Tiny-ImageNet	
Attack↓	Time (s)	Memories (MB)	Time (s)	Memories (MB)	Time (s)	Memories (MB)
Benign	1.62±0.19	76.8	14.11±1.45	125.1	37.92±2.71	233.5
Baseline Attack	1.67±0.25	81.6	14.81±2.10	127.2	38.52±2.19	226.4
DBA	1.57±0.31	81.7	14.91±1.86	124.7	38.74±1.92	248.5
Neurotoxin	3.39±0.66	120.4	27.85±1.74	279.3	76.38±3.46	478.7
FTA	2.04±0.52	86	18.38±1.89	169.1	46.98±2.14	298.4

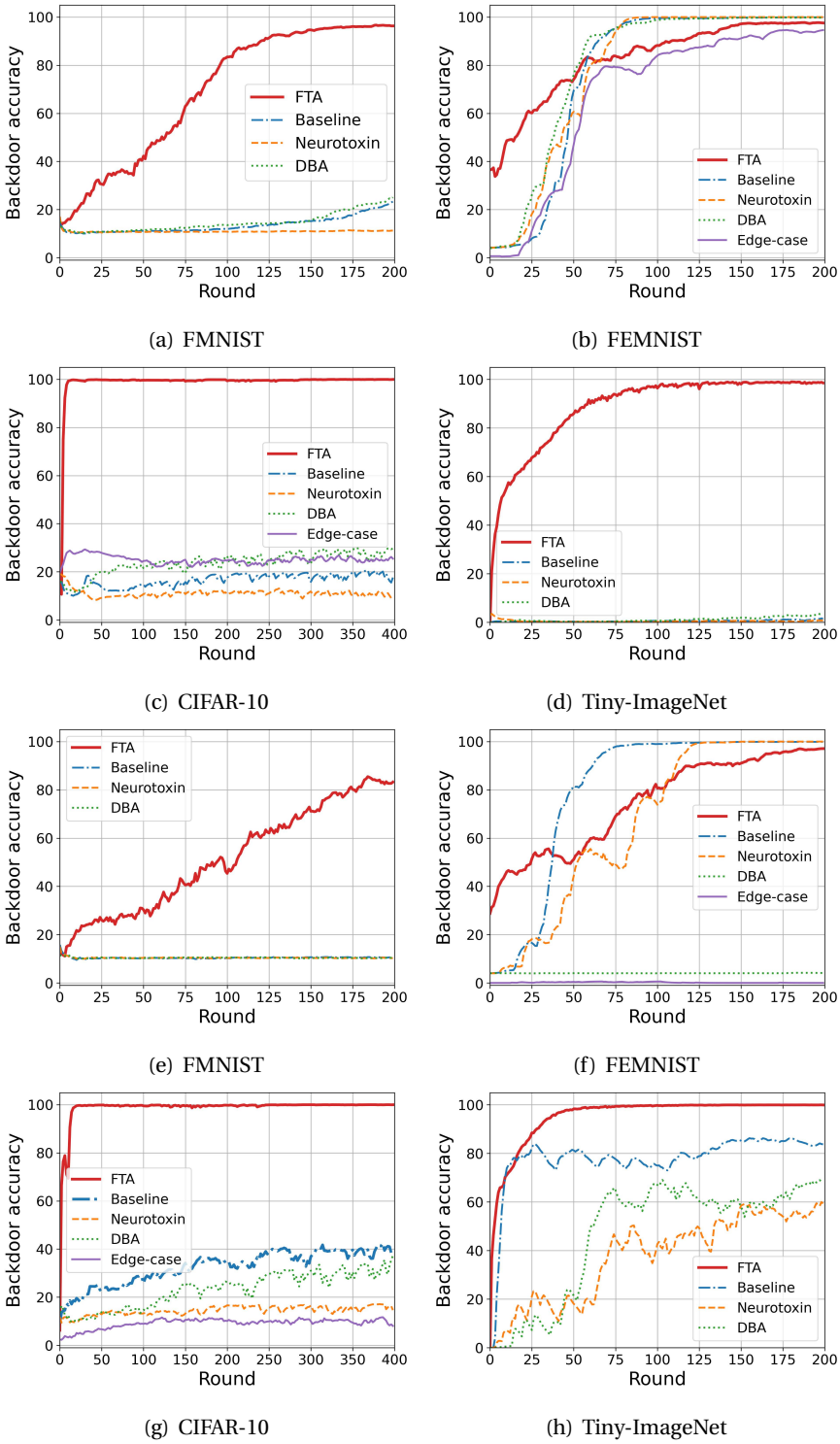
4.5.4. STEALTHINESS AGAINST DEFENSIVE MEASURES

We test the stealthiness **(P1-2)** of FTA and other attacks using 8 SOTA robust FL defenses introduced in Section 4.2.3, such as norm clipping and FLAME, under fixed-frequency scenarios. All four tasks are involved in this defense evaluation. The results, see Figure 4.5 show that FTA can break the listed defenses. Beyond this, we also evaluate different tasks on Multi-Krum, Trimmed-mean, RFA, SignSGD, Foolsgold and SparseFed. FTA maintains its stealthiness and robustness under these defenses.

Resistance to Vector-wise Scaling. We use the norm clipping as the vector-wise scaling defense method, which is regarded as a potent defense and has proven effective in mitigating prior attacks [32]. On the server side, norm clipping is applied on all updates before performing `FedAvg`. Inspired by [26], we utilize the variant of this method in our experiments. As introduced in Section 4.5.2, if we begin the attack from scratch, the norm of benign updates will be unstable and keep fluctuating, making us hard to set a fixed norm bound for all updates. We here filter out the biggest and smallest updates and compute the average norm magnitude based on the rest updates, and set it as the norm bound in current FL iteration.

As shown in Figure 4.5 (a)-(d), this variant of norm clipping can effectively undermine prior attacks in Fashion-MNIST, CIFAR-10, and Tiny-ImageNet. It fails in FEMNIST because benign updates have a larger norm (for example, 1.2 in FEMNIST at round 10, but only 0.3 in Fashion-MNIST), which cannot effectively clip the norm of malicious updates, thus resulting in a higher BA of existing attacks. We see that FTA provides the best BA which is less influenced by clipping than others. FTA only needs a much smaller norm to effectively fool the global model. Although converging a bit slowly in FEMNIST, FTA can finally output a similar performance (above 98%) compared to others.

Resistance to Cluster-based Filtering. The cluster-based filtering defense method is FLAME [30], which has demonstrated its effectiveness in mitigating SOTA attacks against FL. It mainly uses HDBSCAN clustering algorithm based on cosine similarity between all updates and strains the updates with the least similarity compared with other updates. In fig. 4.5 (e)-(h), we see that FLAME can effectively sieve malicious updates of other attacks in Fashion-MNIST and CIFAR-10, but provides relatively weak effectiveness in FEMNIST and Tiny-ImageNet. This is so because data



(a) FMNIST (b) FEMNIST

(c) CIFAR-10 (d) Tiny-ImageNet

(e) FMNIST (f) FEMNIST

(g) CIFAR-10 (h) Tiny-ImageNet

Figure 4.5: Attack stealthiness against defenses. (a)-(d): The variant of norm clipping; (e)-(h): FLAME.

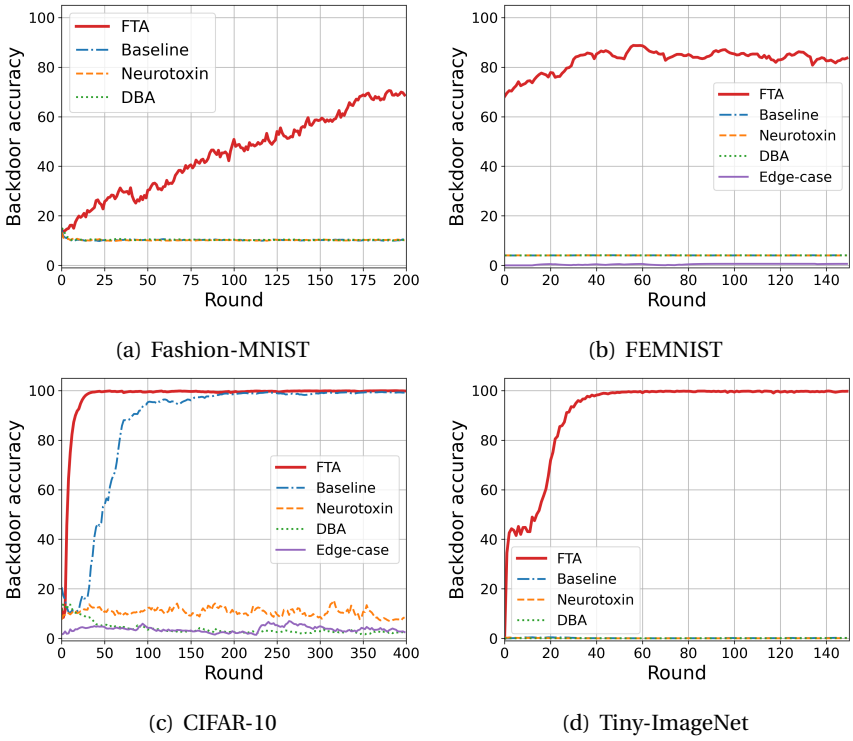
distribution among different agents are fairly in non-i.i.d. manner. Cosine similarity between benign updates is naturally low, making malicious updates possibly evade from the clustering filter.

Similar to the result of Multi-Krum (see Section 4.5.4), FTA achieves >99% BA and finishes the convergence within 50 rounds in CIFAR-10 and Tiny-ImageNet, while delivering an acceptable degradation of accuracy, <20%, in Fashion-MNIST. In FEMNIST, FTA converges slightly slower than the baseline and Neurotoxin but eventually maintains a similar accuracy with only 2% difference. The result proves that FTA enforces malicious updates to have highly cosine-similarity against benign updates due to the same reason in Section 4.5.4, so that it can bypass the defenses based on similarity of updates.

Resistance to vector-wise filtering. Multi-Krum is used as the vector-wise defense method. As described in Section 4.2.3, it calculates the Euclidean distance between all updates and selects $n-f-1$ updates with the smallest Euclidean distances for aggregation. In Figure 4.6, the defense manages to filter out almost all malicious updates of prior attacks and effectively degrade their attacks' performance. In contrast, local update of FTA cannot be easily filtered and thus FTA outperforms others. In CIFAR-10 and Tiny-ImageNet, the attack performance is steady for FTA (nearly 100%) within 40 rounds to converge. In FEMNIST, Multi-Krum only results in a 10% BA degradation for FTA while BAs of others are restricted to 0%. In Fashion-MNIST, Multi-Krum can sieve malicious updates of FTA occasionally, leading to a longer convergence time, but still fails to completely defend the FTA. Malicious updates produced by FTA (which successfully eliminates the anomalies in **P1-2**) are with a similar Euclidean distance compared to benign updates, making them more stealthy than other attacks'.

Resistance to dimension-wise filtering. We choose Trimmed-mean as the representative of dimension-wise filtering. As mentioned in Section 4.2.3, the dimensions of updates are sorted respectively, and the top m highest and smallest updates are removed, and the arithmetic mean of the rest parameters is computed for aggregated updates. In our experiments, m is set as 2 because we assume there is no more than one malicious agent during FL iteration, and setting a higher m can result in lower convergence. As shown in Figure 4.7, Trimmed-mean successfully filters out the compared attacks in Fashion-MNIST and Tiny-ImageNet, and its effects are weakened in CIFAR-10 and FEMNIST. However, FTA survives in all four tasks and performs the best under trimmed-mean. In CIFAR-10, it completes the convergence within 30 rounds and remains 99.9% BA. In Fashion-MNIST and FEMNIST, FTA takes above 50 rounds to fully converge, and the final accuracy manages to reach 96%. The performance of FTA is significantly degraded in Tiny-ImageNet, but still with 30% advantage over other attacks on average. The update of FTA shares a similar weights/biases distribution of benign updates. This ensures our attack to defeat the defenses based on dimension-wise filtering.

Resistance to RFA. In Figure 4.8, FTA provides the best performance among others in Fashion-MNIST, CIFAR-10 and Tiny-ImageNet. In FEMNIST, it converges much faster than prior attacks. Although its accuracy is 8% lower than the baseline in the middle of training, FTA achieves the same performance at the end (of training).



(a) Fashion-MNIST (b) FEMNIST

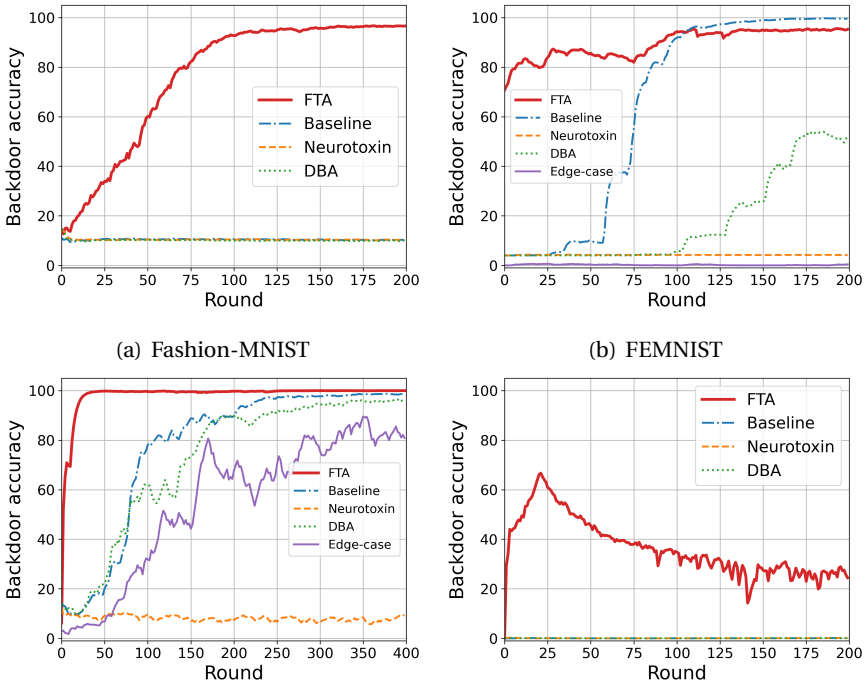
(c) CIFAR-10 (d) Tiny-ImageNet

Figure 4.6: The effectiveness of attack under Multi-Krum in 4 tasks.

Resistance to SignSGD. As shown in Figure 4.9 (a)-(b), SignSGD mitigates prior backdoor attacks with a universal trigger pattern. However, FTA still defeats it and remains 94% and 99% BA on Fashion-MNIST and Tiny-ImageNet, respectively.

Resistance to Foolsgold. From Figure 4.9 (c), we see that Foolsgold hinders the convergence speed of FTA in Fashion-MNIST, which requires FTA to perform extra 25 rounds for convergence. In this sense, FTA still converges much faster than others.

Resistance to sparsification. We choose SparseFed as the representative of the sparsification defense. In Figure 4.9 (d), only Neurotoxin and FTA are capable of breaking through SparseFed on Tiny-ImageNet. The BA of Neurotoxin exhibits fluctuations (between 22% and 36%) throughout the training process, unable to maintain a continuous rise. In contrast, FTA demonstrates the ability to consistently poison the global model and later achieves an impressive accuracy of 90% by round 150. The reason for the above performance difference is that the backdoor task of FTA captures imperceptible perturbations on model parameters, which eliminates the anomalies of poisoning training. The backdoor tasks trained by FTA are more likely to contribute to the same dimensions of gradients as benign updates. Consequently, the top-k filtering mechanism implemented in the server side is ineffective to filter out FTA's backdoor effect.



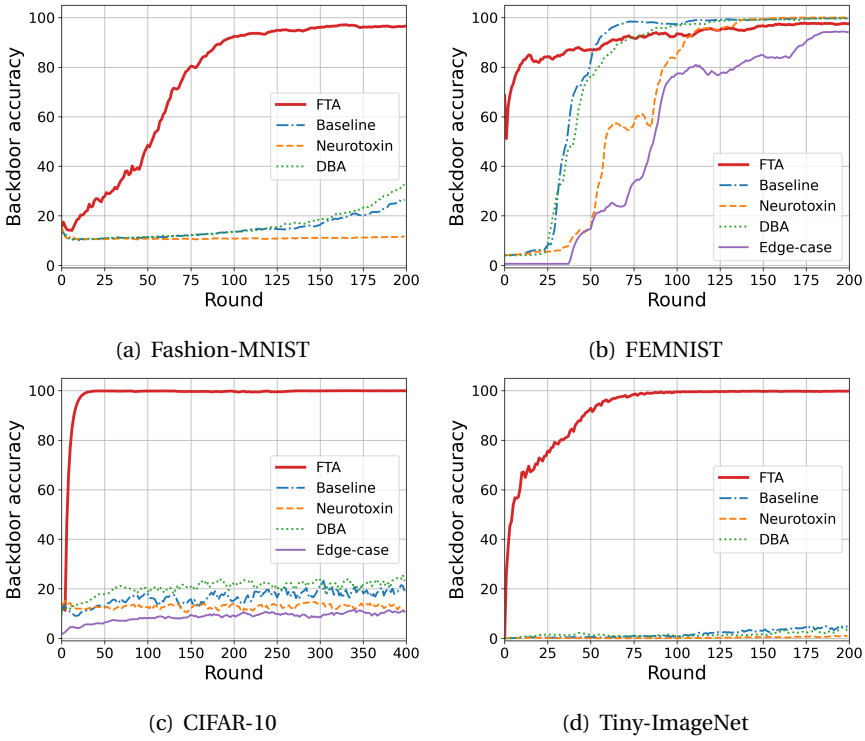
(a) Fashion-MNIST (b) FEMNIST

(c) CIFAR-10 (d) Tiny-ImageNet

Figure 4.7: The effectiveness of attack under Trimmed-mean in 4 tasks.

4.5.5. EXPLANATION VIA FEATURE VISUALIZATION BY T-SNE

We use t-SNE [51] visualization result on CIFAR-10 to illustrate why FTA is more stealthy than the attacks without “flexible" triggers. We select 1,000 images from different classes uniformly and choose another 100 images randomly from the dataset and add triggers to them (in particular, patch-based trigger “square" in baseline method, flexible triggers in FTA). To analyze the hidden features of these samples, we use two global poisoned models injected by baseline attack and FTA respectively. We exploit the output of each sample in the last convolutional layer as the feature representation. Next, we apply dimensionality reduction techniques and cluster the latent representations of these samples using t-SNE. From Figure 4.10 (a)-(b), We see that in the baseline, the distance of clusters between images of the target label “7 and the poisoned images are clearly distinguishable. So the parameters responsible for backdoor routing should do adjustments to map the hidden representations of poisoned images to target label. In FTA, the hidden features of poisoned data overlapped with benign data of target label, which eliminates the anomaly in **feature extraction (P1)**. FTA can reuse the benign routing in FC layers for backdoor tasks, resulting in much less abnormality in **backdoor routing (P2)**, thus the malicious updates can be more similar to benign ones, see Figure 4.10 (c)-(d), producing a



(a) Fashion-MNIST (b) FEMNIST

(c) CIFAR-10 (d) Tiny-ImageNet

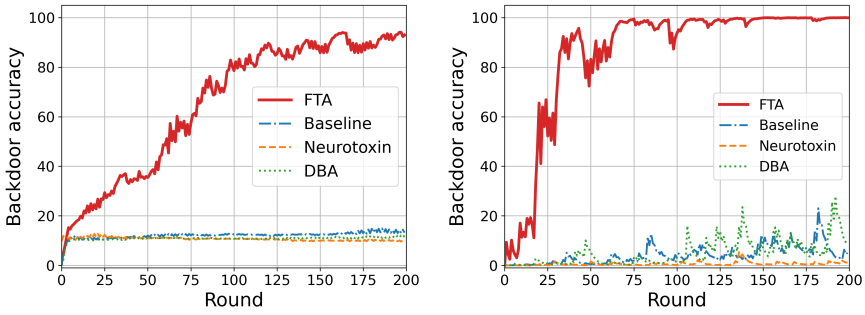
Figure 4.8: The effectiveness of attack under RFA in 4 tasks.

natural parameter stealthiness.

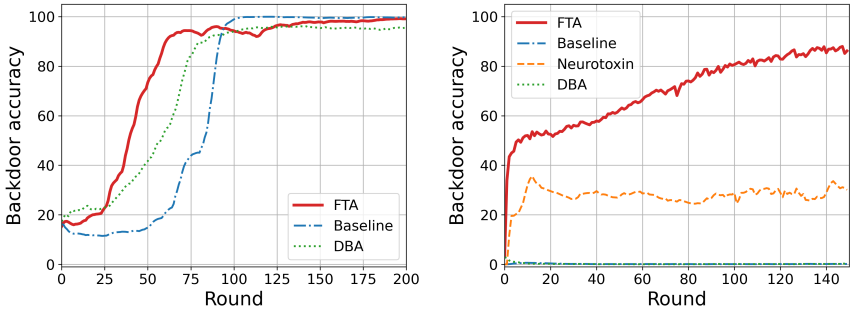
4.5.6. NATURAL STEALTHINESS

We evaluate the natural stealthiness of our backdoor samples by SSIM [52] and LPIPS [53] to indicate that **P3** is well addressed by FTA flexible triggers. For each dataset, we randomly select 500 sample images from test dataset to evaluate the trigger stealthiness. As the SSIM value increases, the poisoned sample looks more stealthy. But for LPIPS, that is the other way round. Table 4.4 shows that FTA achieves excellent stealthiness in all cases. Specifically, SSIM values of FTA are the highest in these datasets, which are close to 1. LPIPS values of FTA are 2-7× improvement to that of baseline attack. Although the baseline attack and Neurotoxin, which uses a universal square pattern, performs well on more complex datasets, using such a patch-based pattern can make the original image look “unnatural".

The benign and poisoned samples with flexible triggers of different sizes generated by FTA are presented in Figure 4.11. For Tiny-ImageNet and CIFAR-10, it is hard for human inspection to immediately identify the triggers, which proves the stealthiness in **P3**. In Fashion-MNIST and FEMNIST, the triggers are easier to distinguish because there is only one channel of the input samples in the datasets. But those flexible



(a) Fashion-MNIST (b) Tiny-ImageNet



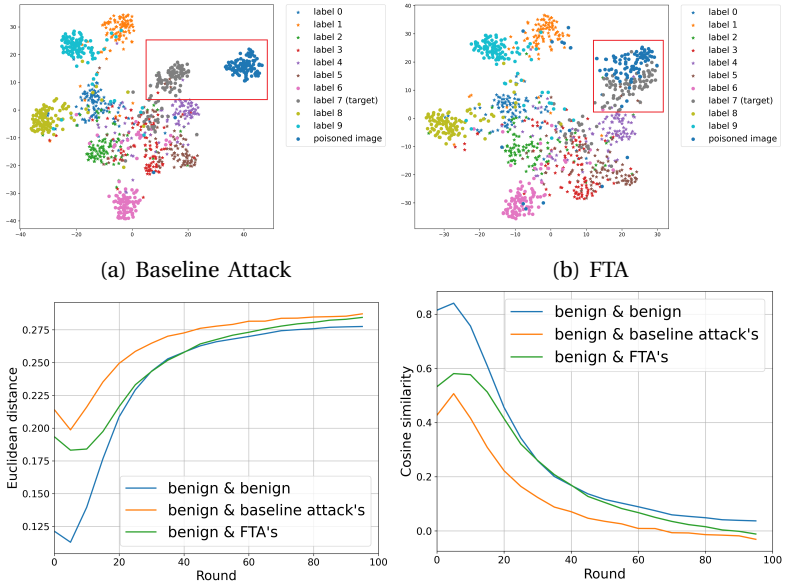
(c) Fashion-MNIST (d) Tiny-ImageNet

Figure 4.9: (a)-(b): The effectiveness of attack under SignSGD in Fashion-MNIST and Tiny-ImageNet. (c): The effectiveness of attack under Foolsgold in Fashion-MNIST. (d): The effectiveness of attack under SparseFed in Tiny-ImageNet.

triggers are still much more stealthy compared to those produced by prior attacks on FL (see Figure 4.2).

Table 4.4: Natural stealthiness of FTA triggers (SSIM↑ and LPIPS↓).

Dataset	Metric	Baseline	DBA	Neurotoxin	Edge-case	FTA(Ours)
Fashion-MNIST	SSIM	0.9376	0.9052	0.9359	-	**0.9967**
	LPIPS	NA	NA	NA	NA	NA
CIFAR-10	SSIM	0.9612	0.9440	0.9638	0.7354	**0.9978**
	LPIPS	0.0058	0.0091	0.0075	0.3171	**0.0008**
Tiny-ImageNet	SSIM	0.9851	0.9734	0.9810	-	**0.9881**
	LPIPS	0.0072	0.0149	0.0086	-	**0.0029**



(a) Baseline Attack (b) FTA

(c) Euclidean distance (d) Cosine similarity

Figure 4.10: (a)-(b): T-SNE visualization of hidden features of input samples in CIFAR-10. The hidden features between poisoned and benign samples of target label is indistinguishable in FTA framework. (c)-(d): Similarity comparison between benign & malicious updates. FTA's malicious updates is more similar to benign updates than the baseline attack's.

4.5.7. ABLATION STUDY IN FTA ATTACK

We here analyze several hyperparameters that are critical for the FTA's performance including trigger size, poison fraction and dataset size of trigger generator.

Trigger size. This size refers to the l_2-norm bound of the trigger generated by the generator, corresponding to ϵ in Equation (4.4). In Figure 4.12 (a)-(d), the trigger size significantly influences the attack performance in all the tasks. The ASRs of FTA drop seriously and eventually reach closely to 0% while we keep decreasing the trigger size, in which evidences can be seen in CIFAR-10, FEMNIST, and Tiny-ImageNet. Additionally, the trigger with a size of 2 in CIFAR-10 and Tiny-ImageNet is indistinguishable from human inspection (see Figure 4.11), while for Fashion-MNIST and FEMNIST (in which images have 2 dimensions), additional noise can be still easily detected. Thus, a balance between visual stealthiness and effectiveness should be considered before performing FTA attack.

Poison fraction. This refers to the fraction of poisoned training samples in the original training dataset of the attacker. Setting a low poison fraction can benefit the attack's stealthiness by having less abnormality in parameters and less influence on benign tasks. But this can slow down the attack effectiveness, as a side effect. Fortunately, we find that FTA can still take effect under a low poison fraction. We set

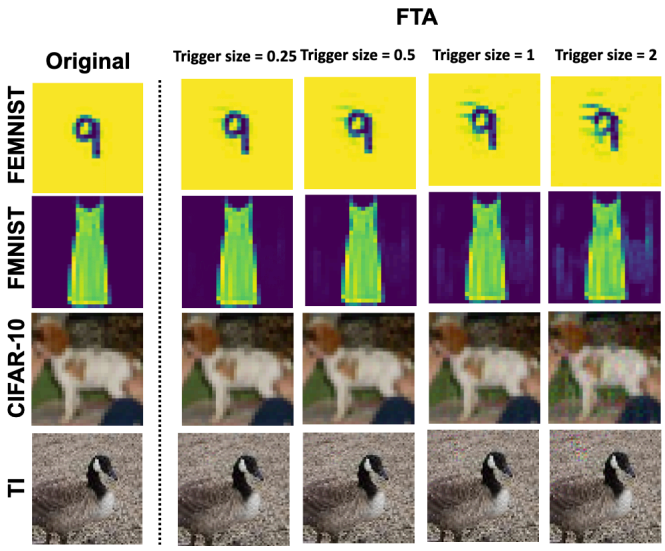
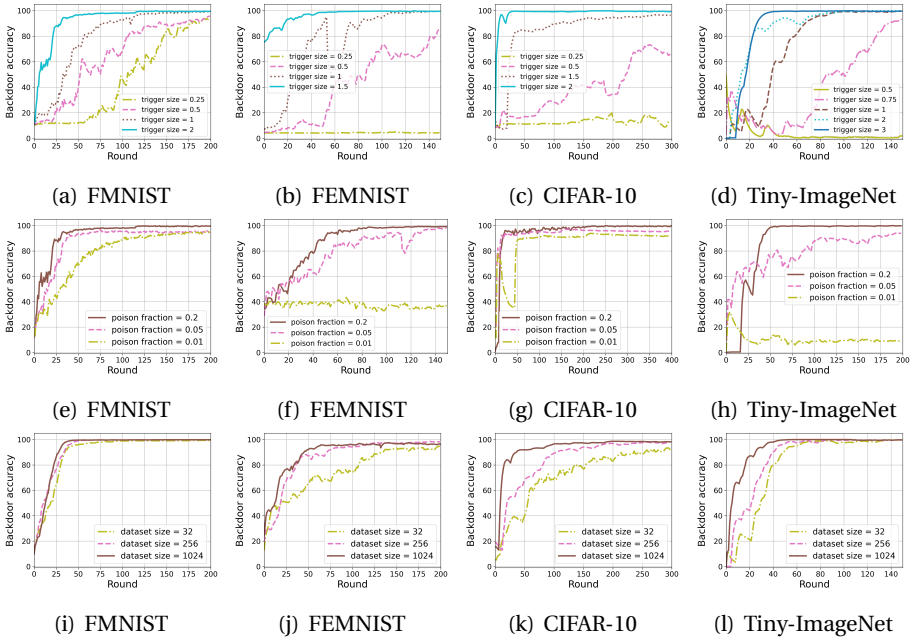


Figure 4.11: Visualization of poisoned images of different trigger sizes.



(a) FMNIST (b) FEMNIST (c) CIFAR-10 (d) Tiny-ImageNet

(e) FMNIST (f) FEMNIST (g) CIFAR-10 (h) Tiny-ImageNet

(i) FMNIST (j) FEMNIST (k) CIFAR-10 (l) Tiny-ImageNet

Figure 4.12: Different hyperparameters on backdoor accuracy. (a)-(d): trigger size; (e)-(h): poison fraction; (i)-(l): dataset size of trigger generator.

the local training batch size to 256 for all the tasks, follow the standard settings of other FL frameworks, and set the poison fraction as 0.1. In Figure 4.12 (e)-(h), FTA is still effective whilst the fraction drops to 0.05. We also find that sensitivities to poison fraction can vary among tasks. In Fashion-MNIST and CIFAR-10, FTA remains its performance even if poison fraction = 0.01, in which only around 3 samples are posoined in each batch. As for FEMNIST and Tiny-ImageNet, under the same rate, the backdoor tasks are dramatically weakened by the benign ones.

Dataset size of trigger generator. Theoretically, if this dataset is small-scale, the trigger generator could not be properly trained, thus resulting in bad quality and further endangering the attack performance. During the training, if the attacker controls multiple agents, it can merge all local datasets into one for generator training. However, in many cases, the attacker can only control relatively limited agents and is provided by a small-scale dataset for training. Recall that in algorithm 5 we use the same dataset for the malicious model and trigger generator training. We set the size of dataset for learning trigger generator to 1024 for all tasks in default. From fig. 4.12 (i)-(l) in section 4.5.7, we see that this concern should not be crucial for FTA. Even if the size of the dataset is only set to 32, FTA can provide a high attack performance. We note that the training process here is somewhat similar to generative adversarial networks, in which we do not require a large amount of samples in the training dataset.

4.6. CONCLUSION AND DISCUSSION

We design an effective and stealthy backdoor attack against FL called FTA by learning an adaptive generator to produce imperceptible and flexible triggers, making poisoned samples have similar hidden features to benign samples with target label. FTA can provide stealthiness and robustness in making hidden features of poisoned samples consistent with benign samples of target label; reducing the abnormality of parameters during backdoor task training; manipulating triggers with imperceptible perturbation for training/testing stage; learning the adaptive trigger generator across different FL rounds to generate flexible triggers with best performance. The empirical experiments demonstrate that FTA can achieve a practical performance to evade SOTA FL defenses.

4.6.1. DISCUSSION

In this work, we concentrate on the computer vision tasks, which have been the focus of numerous existing works [6, 7, 34, 36, 54]. In the future, we intend to expand the scope of this work by applying our design to other real-world applications, such as natural language processing (NLP) and reinforcement learning (RL), as well as other vision tasks, e.g., object detection.

The primary focus of FTA is to achieve stealthiness rather than durability, in contrast to other attacks such as Neurotoxin [8]. Neurotoxin manipulates malicious parameters based on gradients in magnitude, which yields a clear increase in the dissimilarity of parameters and thus harms the stealthiness of the attack. FTA addresses the dissimilarity difference of weights/biases introduced by backdoor

training by using a stealthy and adaptive trigger generator, which makes the hidden features of poisoned samples similar to benign ones. We emphasize that the durability of backdoor attacks on FL is orthogonal to the main focus of this work, and we leave it as an open problem. A possible solution to achieve persistence could be to decelerate the learning rate of malicious agents, as proposed in [10].

Comparison. In addition to the defenses evaluated in this paper, we discuss our attack effectiveness under other defenses below. As depicted in FLDetector [55], in a typical FL scenario where the server does not have a validation dataset that Fltrust [56] requires, the global model remains susceptible to backdoor attacks. However, the stringent demand by FLtrust for an extra validation dataset could not be practical for conventional FL frameworks and applications. Furthermore, Fltrust eliminates backdoor effectiveness based on cosine dissimilarity which is similar to the approach used in FLAME. As shown in fig. 4.10, FTAs malicious updates have less dissimilarity to benign updates than the baseline attacks. Therefore, we can state that FTA can evade Fltrust according to the results obtained under FLAME. DnC [57] primarily focuses on untargeted poisoning attacks rather than backdoor attacks, and its main objective is to reduce the accuracy of FL models. Accordingly, we do not consider it as a “proper" SOTA backdoor defense (to our attack). In particular, DnC is a kind of vector-wise filtering defense. In our experiments, conducted under Multi-krum and RFA, we ascertain that FTA is robust against vector-wise filtering. In conclusion, FTA can also successfully evade DnC, much like Multi-krum and RFA. As for certified defense like Flcert [58], while it is a promising approach to robustness certification, it is not intended to detect and filter out malicious updates in FL. As outlined in Flcert, the certified accuracy of the global model experiences a decline with the increase of malicious agents. Fortunately, FTA can cope with a very challenging threat model, where the attacker is allowed to control merely one malicious agent out of thousands. We thus can achieve a certified accuracy almost on par with the original global model accuracy against Flcert. FLIP [39] only considers static backdoors as potential attacks, i.e. patch-based patterns, whereas FTA can use flexible triggers to break FLIP’s threat model. Using the flexible trigger generator, FTA can produce sample-specific triggers which pose challenges when applying universal trigger inversion method in FLIP’s step 1.

Broader Impacts. Our study exposes the vulnerability of distributed learning systems to practical and stealthy backdoor attacks, which calls for defensive studies to counter generator-based backdoor attacks. In this sense, this work has a positive impact on the future research of machine learning safety. However, there is a concern that our work could be exploited by adversaries in physical world, which potentially brings a negative impact to society.

REFERENCES

[1] I. Dayan, H. R. Roth, A. Zhong, A. Harouni, A. Gentili, A. Z. Abidin, A. Liu, A. B. Costa, B. J. Wood, C.-S. Tsai *et al.* 'Federated learning for predicting clinical outcomes in patients with COVID-19'. In: *Nature medicine* 27.10 (2021), pp. 1735–1743.

[2] A. Nguyen, T. Do, M. Tran, B. X. Nguyen, C. Duong, T. Phan, E. Tjiputra and Q. D. Tran. 'Deep federated learning for autonomous driving'. In: *2022 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2022, pp. 1824–1830.

[3] T. Yang, G. Andrew, H. Eichner, H. Sun, W. Li, N. Kong, D. Ramage and F. Beaufays. *Applied Federated Learning: Improving Google Keyboard Query Suggestions*. 2018. arXiv: 1812.02903 [cs.LG].

[4] M. Paulik, M. Seigel, H. Mason, D. Telaar, J. Kluivers, R. C. van Dalen, C. W. Lau, L. Carlson, F. Granqvist, C. Vandevelde, S. Agarwal, J. Freudiger, A. Byde, A. Bhowmick, G. Kapoor, S. Beaumont, Á. Cahill, D. Hughes, O. Javidbakht, F. Dong, R. Rishi and S. Hung. 'Federated Evaluation and Tuning for On-Device Personalization: System Design & Applications'. In: *CoRR* abs/2102.08503 (2021). arXiv: 2102.08503.

[5] A. N. Bhagoji, S. Chakraborty, P. Mittal and S. Calo. 'Analyzing federated learning through an adversarial lens'. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 634–643.

[6] C. Xie, K. Huang, P.-Y. Chen and B. Li. 'Dba: Distributed backdoor attacks against federated learning'. In: *International Conference on Learning Representations*. 2019.

[7] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee and D. Papailiopoulos. 'Attack of the tails: Yes, you really can backdoor federated learning'. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 16070–16084.

[8] Z. Zhang, A. Panda, L. Song, Y. Yang, M. Mahoney, P. Mittal, R. Kannan and J. Gonzalez. 'Neurotoxin: Durable Backdoors in Federated Learning'. In: *Proceedings of the 39th International Conference on Machine Learning*. 2022, pp. 26429–26446.

[9] H. Li, Q. Ye, H. Hu, J. Li, L. Wang, C. Fang and J. Shi. '3dfed: Adaptive and extensible framework for covert backdoor attack in federated learning'. In: *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2023, pp. 1893–1907.

[10] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin and V. Shmatikov. 'How to backdoor federated learning'. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 2938–2948.

[11] Y. Wang, H. Su, B. Zhang and X. Hu. 'Interpret neural networks by identifying critical data routing paths'. In: *proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 8906–8914.

[12] J. Carnerero-Cano, L. Muñoz-González, P. Spencer and E. C. Lupu. 'Hyperparameter Learning under Data Poisoning: Analysis of the Influence of Regularization via Multiobjective Bilevel Optimization'. In: *arXiv preprint arXiv:2306.01613* (2023).

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio. 'Generative adversarial nets'. In: *Advances in neural information processing systems* 27 (2014).

[14] M. Arjovsky, S. Chintala and L. Bottou. 'Wasserstein Generative Adversarial Networks'. In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by D. Precup and Y. W. Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 214–223.

[15] B. McMahan, E. Moore, D. Ramage, S. Hampson and B. A. y Arcas. 'Communication-efficient learning of deep networks from decentralized data'. In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 1273–1282.

[16] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. McMahan, S. Patel, D. Ramage, A. Segal and K. Seth. 'Practical secure aggregation for federated learning on user-held data. arXiv 2016'. In: *arXiv preprint arXiv:1611.04482* 13 ().

[17] P. Blanchard, E. M. El Mhamdi, R. Guerraoui and J. Stainer. 'Machine learning with adversaries: Byzantine tolerant gradient descent'. In: *Advances in Neural Information Processing Systems* 30 (2017).

[18] B. Zhao, P. Sun, T. Wang and K. Jiang. 'Fedinv: Byzantine-robust federated learning by inversing local model updates'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 8. 2022, pp. 9171–9179.

[19] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar and V. Smith. 'Federated optimization in heterogeneous networks'. In: *Proceedings of Machine learning and systems* 2 (2020), pp. 429–450.

[20] Y. Liu, Y. Kang, X. Zhang, L. Li, Y. Cheng, T. Chen, M. Hong and Q. Yang. 'A communication efficient collaborative learning framework for distributed features'. In: *arXiv preprint arXiv:1912.11187* (2019).

[21] J. Jiang, S. Ji and G. Long. 'Decentralized knowledge acquisition for mobile internet applications'. In: *World Wide Web* 23.5 (2020), pp. 2653–2669.

[22] T. Li, S. Hu, A. Beirami and V. Smith. 'Ditto: Fair and robust federated learning through personalization'. In: *International conference on machine learning*. PMLR. 2021, pp. 6357–6368.

[23] T. Yu, E. Bagdasaryan and V. Shmatikov. 'Salvaging federated learning by local adaptation'. In: *arXiv preprint arXiv:2002.04758* (2020).

[24] D. Yin, Y. Chen, R. Kannan and P. Bartlett. 'Byzantine-robust distributed learning: Towards optimal statistical rates'. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 5650–5659.

[25] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli and A. Anandkumar. 'signSGD: Compressed Optimisation for Non-Convex Problems'. In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by J. Dy and A. Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, 2018, pp. 560–569.

[26] Z. Sun, P. Kairouz, A. T. Suresh and H. B. McMahan. 'Can you really backdoor federated learning?' In: *arXiv preprint arXiv:1911.07963* (2019).

[27] K. Pillutla, S. M. Kakade and Z. Harchaoui. *Robust Aggregation for Federated Learning*. 2022. arXiv: 1912.13445 [stat.ML].

[28] C. Xie, M. Chen, P.-Y. Chen and B. Li. 'Crfl: Certifiably robust federated learning against backdoor attacks'. In: *International Conference on Machine Learning*. PMLR. 2021, pp. 11372–11382.

[29] A. Panda, S. Mahloujifar, A. N. Bhagoji, S. Chakraborty and P. Mittal. 'Sparsefed: Mitigating model poisoning attacks in federated learning with sparsification'. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2022, pp. 7587–7624.

[30] T. D. Nguyen, P. Rieger, H. Chen, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen, A. Mirhoseini, S. Zeitouni, F. Koushanfar, A.-R. Sadeghi and T. Schneider. 'FLAME: Taming Backdoors in Federated Learning'. In: *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 1415–1432. ISBN: 978-1-939133-31-1.

[31] P. Rieger, T. D. Nguyen, M. Miettinen and A.-R. Sadeghi. 'DeepSight: Mitigating backdoor attacks in federated learning through deep model inspection'. In: *NDSS*. 2022.

[32] V. Shejwalkar, A. Houmansadr, P. Kairouz and D. Ramage. 'Back to the Drawing Board: A Critical Evaluation of Poisoning Attacks on Production Federated Learning'. In: *2022 IEEE Symposium on Security and Privacy (SP)* (2021), pp. 1354–1371.

[33] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru and B. Li. 'Manipulating machine learning: Poisoning attacks and countermeasures for regression learning'. In: *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2018, pp. 19–35.

[34] 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. 11966–11976.

[35] K. Doan, Y. Lao and P. Li. 'Backdoor attack with imperceptible input and latent modification'. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 18944–18957.

[36] Z. Zhao, X. Chen, Y. Xuan, Y. Dong, D. Wang and K. Liang. '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, pp. 15213–15222.

[37] Y. Li, Y. Li, B. Wu, L. Li, R. He and S. Lyu. 'Invisible backdoor attack with sample-specific triggers'. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 16463–16472.

[38] N. Zhong, Z. Qian and X. Zhang. 'Imperceptible Backdoor Attack: From Input Space to Feature Representation'. In: *International Joint Conference on Artificial Intelligence*. 2022.

[39] K. Zhang, G. Tao, Q. Xu, S. Cheng, S. An, Y. Liu, S. Feng, G. Shen, P.-Y. Chen, S. Ma and X. Zhang. 'FLIP: A Provable Defense Framework for Backdoor Mitigation in Federated Learning'. In: *The Eleventh International Conference on Learning Representations*. 2023.

[40] O. Ronneberger, P. Fischer and T. Brox. 'U-net: Convolutional networks for biomedical image segmentation'. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer. 2015, pp. 234–241.

[41] K. D. Doan, Y. Lao and P. Li. 'Marksman Backdoor: Backdoor Attacks with Arbitrary Target Class'. In: *arXiv preprint arXiv:2210.09194* (2022).

[42] H. Xiao, K. Rasul and R. Vollgraf. 'Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms'. In: *arXiv preprint arXiv:1708.07747* (2017).

[43] S. Caldas, S. M. K. Duddu, P. Wu, T. Li, J. Konen, H. B. McMahan, V. Smith and A. Talwalkar. 'Leaf: A benchmark for federated settings'. In: *arXiv preprint arXiv:1812.01097* (2018).

[44] A. Krizhevsky, G. Hinton *et al.* 'Learning multiple layers of features from tiny images'. In: (2009).

[45] Y. Le and X. Yang. 'Tiny imagenet visual recognition challenge'. In: *CS 231N* 7.7 (2015), p. 3.

[46] K. Simonyan and A. Zisserman. 'Very Deep Convolutional Networks for Large-Scale Image Recognition'. In: *International Conference on Learning Representations*. 2015.

[47] 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.

[48] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.* 'Pytorch: An imperative style, high-performance deep learning library'. In: *Advances in neural information processing systems* 32 (2019).

[49] C. Fung, C. J. M. Yoon and I. Beschastnikh. 'The Limitations of Federated Learning in Sybil Settings'. In: *Symposium on Research in Attacks, Intrusion, and Defenses*. RAID. 2020.

[50] H. Kusetogullari, A. Yavariabdi, A. Cheddad, H. Grahn and J. Hall. 'ARDIS: a Swedish historical handwritten digit dataset'. In: *Neural Computing and Applications* 32.21 (2020), pp. 16505–16518.

[51] L. Van der Maaten and G. Hinton. 'Visualizing data using t-SNE.' In: *Journal of machine learning research* 9.11 (2008).

[52] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli. 'Image quality assessment: from error visibility to structural similarity'. In: *IEEE transactions on image processing* 13.4 (2004), pp. 600–612.

[53] R. Zhang, P. Isola, A. A. Efros, E. Shechtman and O. Wang. 'The unreasonable effectiveness of deep features as a perceptual metric'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 586–595.

[54] M. S. Ozdayi, M. Kantarcioglu and Y. R. Gel. 'Defending against backdoors in federated learning with robust learning rate'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 10. 2021, pp. 9268–9276.

[55] Z. Zhang, X. Cao and N. Z. Gong. 'FLDetector: Defending Federated Learning Against Model Poisoning Attacks via Detecting Malicious Clients'. In: *KDD*. 2022.

[56] X. Cao, M. Fang, J. Liu and N. Gong. 'FLTrust: Byzantine-robust Federated Learning via Trust Bootstrapping'. In: *Proceedings of NDSS*. 2021.

[57] V. Shejwalkar and A. Houmansadr. 'Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning'. In: *NDSS*. 2021.

[58] X. Cao, Z. Zhang, J. Jia and N. Z. Gong. *FLCert: Provably Secure Federated Learning against Poisoning Attacks*. 2022. arXiv: 2210.00584 [cs.CR].

5

DIFFUSION-BASED PURIFICATION AGAINST ADVERSARIAL ATTACKS

While adversarial training (AT) and purification (AP) offer some defense against adversarial attacks, deep neural networks (DNNs) remain susceptible to increasingly sophisticated attack strategies. This paper introduces Multi-Scale Inpainting Defense (MSID), an AP-based technique that leverages the generative power of Denoised Diffusion Probabilistic Models (DDPMs) to achieve improved robustness. MSID identifies perturbed image regions through multi-scale superpixel segmentation and occlusion analysis, subsequently using DDPMs for inpainting while maintaining visual consistency. This novel approach allows MSID to effectively defend against a wide range of attacks, including those that bypass traditional AT and AP. Experiments on CIFAR-10 and ImageNet demonstrate state-of-the-art performance, with MSID outperforming existing AP methods by up to 5.42% and 10.75% respectively against AutoAttack. MSID also shows improvements against PGD (2.49%) and unseen attacks (36.9%), demonstrating its effectiveness as a robust defense against adversarial attacks.

This chapter is based on the paper “MSID: Multi-Scale Diffusion-Based Inpainting Defense Against Adversarial Attacks" by Popovici, A., Qiao, Y., Liu, D., Smaragdakis, G. and Liang, K., which is under review.

5.1. INTRODUCTION

Deep neural networks (DNNs), despite their impressive image classification capabilities, remain vulnerable to adversarial attacks [1–4]. These attacks introduce imperceptible image perturbations, leading to misclassifications and significant safety concerns [5, 6]. Developing robust defenses is therefore crucial. Two primary defense strategies exist: adversarial training (AT) and adversarial purification (AP). AT [5, 7, 8] enhances robustness by training DNNs on adversarial examples. While effective against known attacks, AT suffers from overfitting, struggles with unseen attacks and image corruptions [9–12], and often degrades standard accuracy while increasing computational complexity [13]. AP offers an alternative, pre-processing images to remove adversarial perturbations before inference [14–16]. Often implemented using generative models [17–22], AP provides a plug-and-play defense against unseen attacks without classifier retraining. However, AP generally shows lower standard accuracy than AT [23, 24]. Creating effective purification models, especially for large-scale datasets, remains challenging due to the inherent trade-off between preserving image semantics and removing perturbations [22].

Recently, diffusion models have emerged as a promising approach for generative AP [25, 26]. Their superior sample quality, exceeding even GANs in image generation [27, 28], make them well-suited. The inherent denoising process aligns with purification, and their stochastic nature offers potential for robust stochastic defenses [29]. These properties make diffusion models a compelling area for improving DNN robustness against adversarial attacks.

While significant progress has been made in defending against adversarial attacks, current methods have limitations. Existing adversarial purification techniques, such as those explored in [19, 22, 30], are vulnerable to color-based attacks. This vulnerability originate from the inherent sensitivity of DDPM models to image colors [19], making them susceptible to even subtle chromatic manipulations introduced by adversaries.

Moreover, AT methods often degrade standard accuracy and struggle with defending unseen attacks [10, 11, 31]. These limitations create a gap in robust image classification: the lack of a defense that simultaneously maintains high standard accuracy, effectively adversarial perturbations and provides robustness against a diverse range of attacks, including color-based attacks and unseen attacks. Therefore, this research develops a new defense that improves resistance to color-based attacks and aims to outperform current state-of-the-art in robust accuracy

MSID performs targeted adversarial perturbation removal through a four-step process: **(1) Hierarchical Perturbation Localization (HPL):** We introduce HPL, a novel method for identifying potentially perturbed image regions at multiple scales. HPL leverages superpixel segmentation at various granularities to capture both coarse and fine-grained image features. This multi-scale approach enables the identification of perturbed regions at different levels of detail, improving the accuracy of subsequent steps. **(2) Occlusion sensitivity map generation:** Occlusion sensitivity maps [32] are generated for each superpixel scale. These maps highlight regions where occlusions (simulated by blurring out superpixels) substantially affect the classifiers output. This identifies areas likely containing adversarial perturbations,

as perturbations in these sensitive regions would cause the greatest misclassification effect. The multi-scale approach allows capturing sensitivity information across different levels of image structure, enhancing the robustness and accuracy of perturbation localization. **(3) Targeted inpainting:** Regions identified as highly sensitive by the occlusion maps are then inpainted using a pre-trained denoising diffusion probabilistic model (DDPM). Unlike classic diffusion models that operate globally on the entire image, inpainting allows for localized restoration, preserving image content while specifically targeting and removing the identified adversarial perturbations. This targeted approach offers greater control and accuracy compared to global denoising, minimizing the risk of altering semantically important image features. **(4) Removing artifacts:** The grid used for inpainting introduces artifacts. Variance Preservation Sampling (VPS) [33] refines the processed image within the diffusion model's latent space, resulting in a natural, artifact-free final image.

Extensive experiments across diverse datasets (CIFAR-10 and ImageNet) and model architectures (ResNet, WideResNet and Vision Transformer (ViT)) demonstrate state-of-the-art performance, surpassing existing AP and AT methods. Our method significantly improves robust accuracy compared to existing AP techniques. Against AutoAttack [34] l_∞, we achieved gains up to 5.42% on CIFAR-10 and 10.75% on ImageNet. Similar improvements were seen against PGD l_∞ (up to 2.49% on CIFAR-10 and 10.75% on ImageNet). Furthermore, against the color-based adversarial attack cADV [35], we achieved 64.84% robust accuracy on Imagenet and 75.19% on CIFAR-10, demonstrating its effectiveness against this attack type. Finally, our approach shows a considerably larger advantage (up to 36.9% improvement) over state-of-the-art adversarial training methods against unseen attacks. This work presents the first application of DDPM inpainting for adversarial purification, offering a novel and effective defense strategy combining Explainable AI (XAI) and AP.

The **main contributions** of this work are as follows:

- This paper, to the best of our knowledge, is the first to apply DDPM inpainting for adversarial purification, enabling targeted removal of perturbations while preserving benign image features, unlike classical diffusion models which operate globally.

- We propose a novel defense technique which combines occlusion sensitivity maps to identify vulnerable regions in adversarial images. This allows precise identification and removal of adversarial perturbations, improving the robsutness compared to methods that lack targeted restoration.

- MSID shows promising adversarial robustness against color-based attacks, a challenging type of attacks needing more research. On CIFAR-10, MSID achieves 75.19% robust accuracy, and on ImageNet, 64.84%.

- Extensive experiments show our method substantially outperforms previous AP methods. MSID achieves gains up to 5.42% (CIFAR-10) and 10.75% (ImageNet) against AutoAttack, with further improvements of 2.49% against PGD and 36.9% against unseen attacks.

5.2. RELATED WORKS

Early work in AP relied heavily on generative adversarial networks (GANs). Samangouei et al. introduced Defense-GAN [36], a system that uses a GAN trained on clean images to identify and remove the adversarial perturbations added to an image before it's classified. A key advantage of Defense-GAN is its model-agnostic nature; it can be used with various DNN architectures and is effective against a wide range of attack methods. Building on this, Song et al. proposed PixelDefend [37], another generative model-based approach. Their insight was that adversarial examples often lie in low-probability regions of the data distribution. PixelDefend cleverly moves these perturbed images towards higher-probability regions, effectively purifying them before classification. This method, too, is model-agnostic and works well with pre-trained classifiers.

Energy-based models (EBMs) have also proven to be a valuable tool for adversarial purification. Du & Mordatch [38] demonstrated the broad applicability of EBMs across diverse tasks, achieving state-of-the-art results in adversarially robust classification. Grathwohl et al. [39] significantly advanced the use of EBMs by reinterpreting standard discriminative classifiers as EBMs, leading to improvements in both robustness and out-of-distribution detection. Subsequently, Hill et al. [20] leveraged Markov Chain Monte Carlo (MCMC) sampling within an EBM framework to purify adversarial examples, achieving considerable results even against the most advanced attacks. Srinivasan et al. [14] presented MALADE, which uses the Metropolis-adjusted Langevin algorithm (MALA) to effectively project adversarial samples onto the correct class manifold, also achieving state-of-the-art performance. Yoon et al. [21] further enhanced the efficiency of EBM-based purification by incorporating Denoising Score Matching, dramatically reducing the number of computationally expensive MCMC steps needed.

More recently, diffusion models have emerged as a powerful new tool for AP. Nie et al. proposed DiffPure [19], which employs a forward and reverse diffusion process to effectively remove adversarial noise. Wang et al. [22] integrated guidance into the denoising process of a Denoised Diffusion Probabilistic Model (DDPM) for purifying images. Unlike existing methods, AGDM [40] introduces a novel adversarial guidance that incorporates semantic information without directly involving adversarial perturbations. This guidance is implemented via an auxiliary neural network trained adversarially, focusing on distances in the latent representation space rather than at the pixel level. Chen et al. [41] further advanced this area with the Robust Diffusion Classifier (RDC), a generative classifier built upon pre-trained diffusion models. RDC maximizes data likelihood and employs Bayes' theorem for robust classification without relying on specific adversarial attack training, also incorporating a new multi-head diffusion backbone and efficient sampling strategies to reduce computational cost. Finally, Lin et al. [30] proposed AToP (Adversarial Training on Purification), a novel hybrid approach that combines adversarial training and purification to address the individual limitations of each method, aiming for improved robustness and better generalization to unseen attacks.

5.3. PRELIMINARY

5.3.1. ADVERSARIAL ATTACKS

We focus on adversarial attacks against deep neural networks, denoted by $f_\theta(x)$, where θ represents the model's parameters and $x \in \mathcal{X}$ is the input data, belonging to the input space $\mathcal{X} \subset \mathbb{R}^d$. The output of the DNN is a prediction vector $f_\theta(x) \in \mathbb{R}^K$, where K is the number of classes. We assume a classification setting where the predicted class is given by $\arg\max_k f_\theta(x)_k$.

Figure 5.1: Illustration of adversarial example generation. The original image x is perturbed by δ to create the adversarial example x', which is misclassified by the DNN f_θ.

Adversarial Examples. An adversarial example is a carefully crafted perturbation, δ, of a benign input x, resulting in $x' = x + \delta$. This perturbation is designed to be imperceptible to humans, thus constrained in magnitude, $|\delta|_p \leq \epsilon$, while simultaneously causing the DNN, f_θ, to misclassify the input (as visualized in Figure 5.1). Formally, an adversarial example x' satisfies:

$$f_\theta(x') \neq f_\theta(x) \quad \text{(Misclassification)},$$
$$\|\delta\|_p < \epsilon \quad \text{(Perceptual similarity)},$$

where $|\cdot|_p$ denotes the ℓ_p-norm, commonly ℓ_∞ (maximum perturbation per feature), ℓ_2 (Euclidean distance), or ℓ_1 (Manhattan distance), and ϵ represents the maximum permissible perturbation magnitude (the perturbation budget). Effectively, generating an adversarial example involves maximizing the loss function of the classifier, f_θ (or similarly, minimizing the classifier's confidence in the true label):

$$\delta = \arg\max_{||\delta|| \leq \epsilon} \mathcal{L}(f_\theta(x+\delta), y), \tag{5.1}$$

where $\mathcal{L}$ is the loss function and y is the true label of x.

5.3.2. ADVERSARIAL TRAINING AND ADVERSARIAL PURIFICATION

Adversarial attacks exploit the sensitivity of the models by crafting carefully perturbed inputs, $x' = x + \delta$, that induce misclassification [5]. These perturbations, δ, are typically constrained in magnitude $\|\delta\| \leq \epsilon$ to maintain perceptual similarity to the original input x, yet effectively maximize the loss function of the classifier f_θ:

$$\delta = \arg\max_{\|\delta\| \leq \epsilon} \mathcal{L}(f_\theta(x+\delta), y), \tag{5.2}$$

where ϵ represents the maximum permissible perturbation scale, often measured using norms like l_∞ or l_2. Several defense mechanisms [8, 42] have been proposed to counter these attacks, with AT and AP being prominent approaches.

Adversarial training (AT) enhances the robustness of f_θ by explicitly incorporating adversarial examples into the training process. This involves solving a min-max optimization problem:

$$\min_\theta \mathbb{E}_{p_{\text{data}}(x,y)} \left[\max_{\|\delta\| \leq \epsilon} \mathscr{L}(f_\theta(x+\delta), y) \right]. \tag{5.3}$$

Essentially, AT aims to minimize the classifier's loss on the worst-case adversarial perturbations within the ϵ-ball around each training sample. While effective, this approach incurs a significant computational overhead due to the inner maximization step and can sometimes lead to a trade-off between robustness and standard data accuracy.

Adversarial purification (AP) offers an alternative defense strategy by introducing a separate purification model g_γ. This model acts as a pre-processing step, transforming the potentially adversarial input x' before it is fed to the classifier f_θ. The goal is to mitigate the impact of the adversarial perturbation δ, ideally resulting in the same classification output as the clean input:

$$f_\theta(g_\gamma(x+\delta)) = f_\theta(x). \tag{5.4}$$

Notably, AP does not necessitate perfect reconstruction of the original input $\big(g_\gamma(x+\delta) \neq x\big)$. Instead, it focuses on removing the adversarial noise sufficient for correct classification. By focusing on removing classification-disrupting noise, rather than perfectly reconstructing the input, AP becomes a versatile tool applicable to a wide range of classifiers. This characteristic allows AP to function as a plug-and-play module, compatible with various classifiers and often implemented using pre-trained generative models for g_θ. The effectiveness of AP, however, relies heavily on the purifier's ability to distinguish and neutralize adversarial perturbations without excessively affecting the underlying semantic content of the input.

5.3.3. DIFFUSION MODELS

Diffusion models [25, 26] have emerged as a powerful class of generative models, demonstrating remarkable capabilities in synthesizing high-quality images through an iterative denoising process applied to samples from a known distribution. Their underlying mechanism involves two key processes: a forward diffusion process and a reverse denoising process.

Forward diffusion process. It, also known as the diffusion process, gradually adds Gaussian noise to the data distribution $p(\mathbf{x}_0)$ over T time steps. This process can be defined by a Markov chain with transition probabilities $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ given by

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}), \tag{5.5}$$

where $\beta_t \in (0,1)$ are variance schedules controlling the amount of noise added at each time step t, and $\mathbf{I}$ is the identity matrix. Commonly used schedules include

linear, cosine, and sigmoid schedules. The schedule can be learned as well. This formulation allows us to sample $\mathbf{x}_t$ at any time step t directly from $\mathbf{x}_0$ using:

$$p(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}), \tag{5.6}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$ and $\alpha_t = 1 - \beta_t$. As t increases, $\mathbf{x}_t$ becomes increasingly noisy and converges to a standard Gaussian distribution when $T \to \infty$.

Backward diffusion process. The backward diffusion process, also known as the reverse process, aims to learn the reverse of the forward diffusion. It starts from a sample $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$ and iteratively removes noise to generate a sample from the data distribution $p(\mathbf{x}_0)$. The reverse process is also a Markov chain defined by

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\gamma(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\gamma(\mathbf{x}_t, t)), \tag{5.7}$$

where $\boldsymbol{\mu}_\gamma(\mathbf{x}_t, t)$ and $\boldsymbol{\Sigma}_\gamma(\mathbf{x}_t, t)$ are the learned mean and covariance, respectively, parameterized by a neural network with parameters γ. This neural network is typically a U-Net architecture [43, 44]. In practice, the covariance is often fixed to a time-dependent constant derived from the forward process variance schedule, and only the mean is learned. The goal of training a diffusion model is to learn the parameters γ such that the reverse process faithfully approximates the true posterior $p(\mathbf{x}_{t-1}|\mathbf{x}_t)$. This is typically done by minimizing a variational bound on the negative log-likelihood.

The inherent denoising capability of diffusion models presents a compelling opportunity for their application in adversarial purification. By interpreting adversarial perturbations as a form of noise added to the clean input, diffusion models can be leveraged to denoise and purify adversarial examples.

5.4. METHOD

5.4.1. OVERVIEW OF PROPOSED FRAMEWORK

Our defense method comprises two main stages (see Figure 5.2): (1) generating occlusion sensitivity maps and (2) restoring image via an inpainting diffusion model. The first stage leverages the observation that the same image region, when considered at different scales, can have distinct impacts on the class prediction score. (see Figure 5.2 (b)). Analyzing the occlusion sensitivity map of an adversarial image reveals the region most responsible for misclassification. The second stage then inpaints these sensitive, presumably perturbed, regions using a DDPM-based inpainting model. This simultaneously removes the adversarial perturbations while preserving image similarity.

5.4.2. GENERATE OCCLUSION SENSITIVITY MAPS

Explainable AI (XAI) research primarily aims to reveal the underlying reasoning in machine learning models [45]. Numerous methods have been introduced, including attribution techniques [46], concept-based approaches [47, 48], and global analysis tools [49, 50]. Our approach applied a model-agnostic technique that focuses on

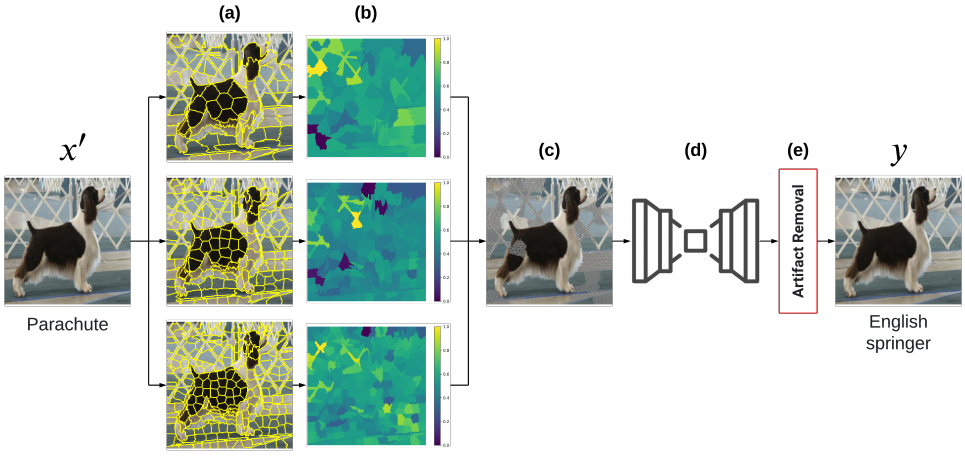


Figure 5.2: The proposed Multi-Scale Superpixel Inpainting for Defense (MSID) method removes targeted adversarial perturbations from images in four steps. Step **(a)**: the adversarial example x' undergoes multi-scale superpixel segmentation, capturing both coarse and fine details to precisely identify potentially perturbed regions. Step **(b)**: occlusion sensitivity maps are generated at each superpixel scale. This involves blurring individual superpixels and measuring the resulting impact on classifier output, effectively highlighting potentially perturbed areas. Step **(c)**: a mask M is created to identify highly sensitive regions based on these sensitivity maps, which are then restored using a pre-trained DDPM (Step **(d)**). This targeted inpainting removes the perturbations while preserving uncorrupted image features. Step **(e)**: Variance Preservation Sampling (VPS) is applied to the restored image to mitigate grid-based inpainting artifacts, ensuring an artifact-free final output y.

occluding specific features and observing the resulting changes in model predictions [32]. Two key factors determine the accuracy of an occlusion sensitivity map: pixel grouping and the choice of imputer for occluded features. These choices are critical because they directly affect the map's ability to identify true sensitive regions and therefore, the effectiveness of any defense built upon it. Incorrect choices can lead to a weak defense. To ensure accurate identification of sensitive regions, we use superpixels for grouping, which allows us to capture meaningful image features. Furthermore, we use Gaussian blurring technique to impute occluded superpixels. Below we detail the rationale behind these choices and explain why they are critical for building a robust defense.

Pixel Grouping: For images $x \in \mathbb{R}^{w \times h \times n}$ with width w, height h, and n color channels, individual pixels often represent redundant information due to their proximity to one another [51]. Therefore, it is more effective to to treat each cluster of pixels (superpixels) as a singular feature. Superpixels are created by segmenting

the entire image into separate patches $N = \{1, 2, \ldots, n_{superpixels}\}$, which reduces the computational load, because $n_{superpixels} < w \cdot h \cdot n$. For our method, we employed the SLIC algorithm [52], as it typically generates superpixels that adhere to local gradients in the image, grouping contextually related pixels and thus enhancing the interpretability of sensitivity maps.

Feature Occlusion: In creating the occluded prediction $f_c(x_S)$ (where x_S denotes the part of the image X restricted to the feature subset S, identified using SLIC), it is generally infeasible to exclude features $\overline{S}$ (the complement of S) entirely. Instead, their influence on the models predictions must be minimized. The only model-agnostic approach is to generate occluded samples $(x_S, X_{\overline{S}})$, where $X_{\overline{S}}$ represents artificially generated values provided by an imputer q. The occluded model prediction is then expressed as:

$$f_c(x_S) = \sum_{X_{\overline{S}} \sim q} f_c(x_S, X_{\overline{S}}). \tag{5.8}$$

Here, we used the marginal distribution $q = p(X_{\overline{S}})$ to decouple S and $\overline{S}$. Our chosen imputer, a Gaussian blurring technique, replaces occluded regions with a blurred effect, preserving the broader structure of the image while eliminating finer details. Given a binary mask M, the blurred image X' is defined as:

$$X' = \text{Blur}(X_{\overline{S}}, M, \sigma). \tag{5.9}$$

The Gaussian blur, controlled by its standard deviation σ, is chosen as our imputer because blurring hides features, preserving context, unlike erasure which introduces discontinuities in the image. This helps minimize out-of-distribution (OOD) artifacts by maintaining contextual similarity in occluded areas [53]. Furthermore, blurring can bring adversarial examples closer to their groundtruth labels, potentially alleviating adversarial properties [54].

Multi-Scale Superpixel Segmentation: Traditional occlusion sensitivity is calculated by occluding patches over the image and observing how much the classifiers output changes [51]. However, this does not align with the contextual image features. In order to find the region that contributes the most to the classification and presumably contains the adversarial noise that triggers the misclassification, our method uses superpixels at three various scales (fine, medium and coarse). Occluding at multiple scales reveals how both fine details and broader image structures contribute to the classification. For instance, fine-scale occlusion might highlight subtle texture differences essential for differentiating dog breeds, while coarse-scale occlusion might show the importance of overall dog shape.

Map Fusion: We combine sensitivity maps from multiple scales using a weighted average, prioritizing coarser maps to highlight major regions while incorporating finer details. Coarser maps, with their broader view, provide the overall structure, while finer maps add the necessary precision. The fused sensitivity map is generated as follows: First, distinct regions are identified within the coarsest-scale sensitivity map. Then, for each identified region, a weighted average of the corresponding areas in the finer-scale maps is calculated (see Equation (5.10)). The weights w_n, where $n \in 1, 2, 3$, correspond to the relative importance of scales 1, 2, and 3, respectively.

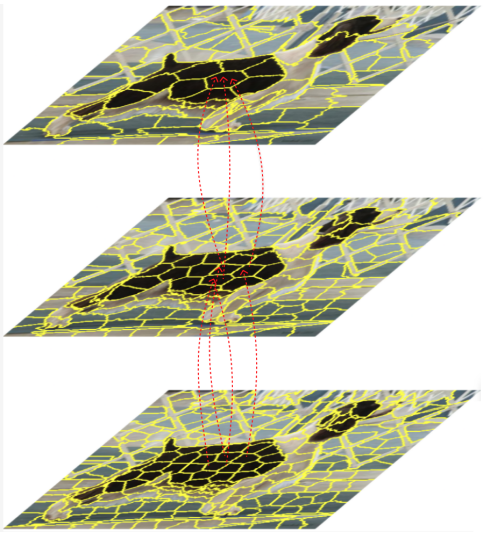


Figure 5.3: Visualization of the multi-scale sensitivity map fusion process. The arrows indicate how regions in finer-scale maps correspond to coarser maps.

The terms F_n denote the sets of elements associated with the n-th fine scale, while $|F_n|$ indicates the number of elements in these sets. Within the summations, S_j represents the score corresponding to each superpixel j in the respective sets. Finally, S_1 is the score for coarse scale. This iterative masking and averaging process yields a smoothed sensitivity map, highlighting the most important areas with increased accuracy (see Equation (5.10)). Figure 5.3 illustrates this fusion process, showing how information from different scales is combined.

$$S_{\text{combined}} = w_3 \cdot \left(\frac{1}{|F_3|} \sum_{f \in F_3} S_j \right) + w_2 \cdot \left(\frac{1}{|F_2|} \sum_{f \in F_2} S_j \right) + w_1 \cdot S_1 \tag{5.10}$$

Our multi-scale approach offers a significant advantage by integrating hierarchical spatial information, providing a deeper understanding of regional significance at different levels. This results in a refined heatmap that accurately reflects important image areas without reducing detail, thus offering richer insights into the classifier's decision-making process.

Clustering: To effectively identify and target the most important image regions for classification, we employ k-means clustering [55] on the superpixel sensitivity scores. This approach allows us to move beyond simply selecting the top-k most sensitive superpixels and instead identify clusters that collectively contribute significantly to the model's decision. By clustering based on sensitivity, we capture the inherent grouping of important image areas, potentially encompassing regions that might be missed by a purely rank-based selection. Finally, we obtain the mask M by clustering the sensitivity regions and selecting the top k most sensitive. This mask is then used for the inpainting process.

5.4.3. IMAGE RESTORATION VIA DDPM-BASED INPAINTING

Once the most sensitive regions of the image are identified and masked, MSID uses the recently proposed DiffPIR framework [56] to perform inpainting. DiffPIR leverages the power of DDPMs within a plug-and-play image restoration framework. It addresses the image restoration problem formulated as:

$$\mathbf{x} = \arg\min_{\mathbf{x}} \frac{1}{2\sigma_n^2} \|\mathbf{y} - \mathscr{H}(\mathbf{x})\|^2 + \lambda \mathscr{P}(\mathbf{x}), \tag{5.11}$$

where z represents the degraded image, $\mathscr{H}(\cdot)$ is the degradation operator (in our case masking the most important region), σ_n^2 is the noise variance, and $\mathscr{P}(\cdot)$ is the prior probability distribution with regularization parameter λ. DiffPIR tackles this optimization problem Equation (5.11) by decoupling the data term and the prior term. This decoupling is achieved using the Half-Quadratic Splitting (HQS) algorithm [57], which iteratively solves two subproblems through the introduction of an auxiliary variable z as follows:

$$z_k = \arg\min_{z} \frac{1}{2(\lambda/\mu)^2} \|z - x_k\|^2 + \mathscr{P}(z), \tag{5.12}$$

$$x_{k-1} = \arg\min_{x} \|y - \mathscr{H}(x)\|^2 + \mu\sigma_n^2 \|x - z_k\|^2, \tag{5.13}$$

Prior Subproblem: This subproblem (see Equation (5.12)) involves enforcing the prior distribution of the image, which is modeled by a denoising diffusion model. The prior term is solved using a Gaussian denoising operator, where the goal is to recover the clean image from the noisy image.

Data Subproblem: This subproblem (see Equation (5.13)) enforces consistency with the observed data y given the degradation operator $\mathscr{H}(\cdot)$. For inpainting, $\mathscr{H}(\cdot)$ represents a mask that hides the perturbation region, and the data subproblem aims to find an image that matches the unmasked pixels while respecting the prior imposed by the DDPM.

To establish a connection between Equation (5.12) and the diffusion process, consider the objective of recovering a noise-free image $\mathbf{z}_k$ from a noisy image $\mathbf{x}_t$ with a noise level $\bar{\sigma}_t$ defined as $\sqrt{\frac{\lambda}{\mu}} = \bar{\sigma}_t$. Given the noise schedule $\{\beta_t\}$ and the hyperparameter λ, which acts as a guidance scaling parameter, much like in classifier-free diffusion models, the value of $\bar{\sigma}_t$ is known. Equation (5.12) may be interpreted as a proximal operator. Recognizing that the gradient of the negative log-likelihood of the image prior $\mathscr{P}(\mathbf{x})$ is equivalent to the negative score function $-s_\theta(\mathbf{x})$, we can reformulate Equation (5.12) as:

$$z_k \approx x_k + \frac{1 - \overline{a_t}}{\bar{a}_t} s_\theta(x_k). \tag{5.14}$$

This implies that z_k represents the estimated clean image x_0^t obtained using the "Variance Exploding" Stochastic Differential Equation (SDE) formulation of diffusion models, where $s_\theta(x_k)$ denotes the score function parameterizing the diffusion mode. For clarity, Equation (5.12) and Equation (5.13) can be expressed as a three-step

process:

$$x_0^{(t)} = \arg\min_z \frac{1}{2\bar{\sigma}_t^2}\|z - x_t\|^2 + P(z), \tag{5.15}$$

$$\hat{x}_0^{(t)} = \arg\min_x \|y - \mathscr{H}(x)\|^2 + \rho_t \|x - x_0^{(t)}\|^2, \tag{5.16}$$

$$x_{t-1} \leftarrow \hat{x}_0^{(t)}, \tag{5.17}$$

where $\rho_t = \lambda\left(\frac{\sigma_n}{\bar{\sigma}_t}\right)^2$. Equation (5.16) represents the denoising step leveraging the diffusion prior, aiming to find the most probable noise-free image $x_0^{(t)}$ given the noisy input x_t. Subsequently, Equation (5.15) functions as the data term, refining the denoised image $x_0^{(t)}$ by incorporating the observed data y and the forward operator $H(\cdot)$. Finally, Equation (5.17) updates the image estimate for the next iteration of the algorithm.

DiffPIR [56] observes that the noise term may not provide sufficient perturbation. Therefore, it introduces a hyperparameter ζ to control noise injection. This modification leads to the explicit formulation presented as follows:

$$\sqrt{\bar{\alpha}_{t-1}}\hat{x}_0^{(t)} + \sqrt{1-\bar{\alpha}_{t-1}}(\sqrt{1-\zeta}\hat{\epsilon} + \sqrt{\zeta}\epsilon_t). \tag{5.18}$$

ζ governs the variance of the noise injected at each step. In particular, the sampling strategy becomes deterministic when ζ is set to 0. Algorithm 6 provides a comprehensive outline of the DiffPIR algorithm.

Algorithm 6: DiffPIR

Require: s_θ, T, y, σ_n, $\{\bar{\sigma}_t\}_{t=1}^T$, ζ, λ
Ensure: The restored image x_0
Initialize $x_T \sim \mathcal{N}(0, I)$, pre-calculate $\rho_t \triangleq \lambda\frac{\sigma_n^2}{\bar{\sigma}_t^2}$
for $t = T$ *to* 1 **do**
  $x_0^{(t)} = \sqrt{\frac{1}{\bar{\alpha}_t}}(x_t + (1-\bar{\alpha}_t)s_\theta(x_t, t))$ // Predict $\hat{z}_0$ with score model as denoiser
  $\hat{x}_0^{(t)} = \arg\min_x \|y - H(x)\|^2 + \rho_t\|x - x_0^{(t)}\|^2$ // Solving data proximal subproblem
  $\hat{\epsilon} = \sqrt{\frac{1}{1-\bar{\alpha}_t}}(x_t - \sqrt{\bar{\alpha}_t}\hat{x}_0^{(t)})$ // Calculate effective $\hat{\epsilon}(x_t, y)$
  $\epsilon_t \sim \mathcal{N}(0, I)$
  $x_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{x}_0^{(t)} + \sqrt{1-\bar{\alpha}_{t-1}}(\sqrt{1-\zeta}\hat{\epsilon} + \sqrt{\zeta}\epsilon_t)$ // Reverse diffusion sampling
end
return x_0

When it comes to inpainting, DiffPIR demonstrates significant advantages over other diffusion-based methods. Its ability to handle arbitrary degradation operators, as opposed to DDRM's [58] linear operators, makes it particularly well-suited for inpainting tasks where the missing regions can be represented by complex masks. Furthermore, DiffPIR is faster compared to Diffusion Posterior Sampling (DPS) [59],

especially when dealing with large or complex missing areas. The enhanced speed does not come at the cost of quality, DiffPIR consistently produces high-fidelity inpainted images, even with fewer sampling steps, outperforming the reconstruction accuracy of DPS in such scenarios.

Contextual cues: Our defense strategy is built on the simple observation (see Figure 5.2 **(b)**) that different parts of an image hold varying levels of importance when a model makes a decision. Some areas are more "sensitive" than others, meaning they have a greater impact on the model's decision. By identifying and masking these sensitive regions, we aim to neutralize adversarial perturbations while preserving the overall structure of the image. However, inpainting models rely on the surrounding context to fill in missing information, and removing entire regions can strip them of the context they need to accurately reconstruct the original image. To address this, we use a strategy of selectively unmasking certain pixels within the masked region. Providing the inpainting model with contextual information allows it to better understand the underlying structure and content of the masked area, leading to more accurate and realistic inpainting results. To aid this process, we use a square grid with some randomness.

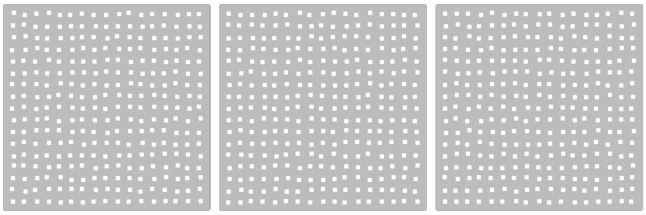


Figure 5.4: Examples of jittered grids for providing contextual cues during inpainting. Each subfigure shows a different realization of the jittered grid applied to the same masked region. The variation in grid point placement helps avoid deterministic patterns and provides more diverse contextual information.

Each point on this grid represents the top-left corner of a square, and we introduce a slight "jitter" to each point's position. This creates a more organic arrangement, subtly shifting each point from its original grid location. Figure 5.4 shows several examples of this jittered grid approach, illustrating the variation in unmasked pixel placement. This approach helps to avoid deterministic patterns and ensures a more realistic distribution of sampling points, ultimately contributing to the effectiveness of our inpainting method.

Removing artifacts: Our AP defense leverages a masking and inpainting approach. To ensure the inpainting model retains sufficient context for accurate reconstruction, we unmask a set of pixels within the masked sensitive regions using a jittered square grid. While this provides contextual cues, it introduces a new challenge: the unmasked pixels themselves become artifacts within the inpainted image. To address this, we introduce a final refinement step using Variance Preservation

Sampling (VPS) technique [33]. Figure 5.5 illustrates the effect of VPS on removing the grid artifacts from the inpainted image. VPS operates by iteratively refining the image representation within the diffusion models latent space, guiding it towards a high probability region that corresponds to clean, artifact-free images. This process effectively eliminates the unwanted unmasked pixels while preserving the integrity of the underlying image structure and the quality achieved in the initial inpainting step.

The core idea is to guide a degraded image towards a "clean" state by ensuring it follows the learned distribution of the pre-trained model, while remaining similar to the original degraded input. This is achieved through a two-stage process: ODE inversion for faithfulness and VPS for restoration.

First, given a degraded image y, we approximate invertibility of the diffusion model's ODE sampling process. Specifically, Denoising Diffusion Implicit Models (DDIM) inversion is used to find a corresponding latent representation y_τ:

$$y_\tau = \text{DDIM}^{-1}(y) \tag{5.19}$$

The parameter τ controls the strength of this inversion. While y_τ effectively encodes the degraded image, it typically resides in a low-probability region of the diffusion model's latent space, making direct generation of a high-quality restoration unlikely. Subsequently, VPS refines the latent y_τ by iteratively guiding it towards a nearby high-probability region, representing the distribution of clean images learnt by the diffusion model. This iterative refinement consists of two steps at each timestep $t \in [\tau, 0)$: First, the latent is updated M times using a combination of the gradient of the log-probability density (computed using the pre-trained diffusion model) as follow:

$$y_t^m = y_t^{m-1} + \eta_l \nabla \log p_t\left(y_t^{m-1}\right) + \eta_g \epsilon^m \tag{5.20}$$

such that η_l and η_g are bound by the constraint:

$$\eta_l = \gamma(1-\bar{\alpha}_t), \quad \eta_g = \sqrt{\gamma(2-\gamma)}\sqrt{1-\bar{\alpha}_t}. \tag{5.21}$$

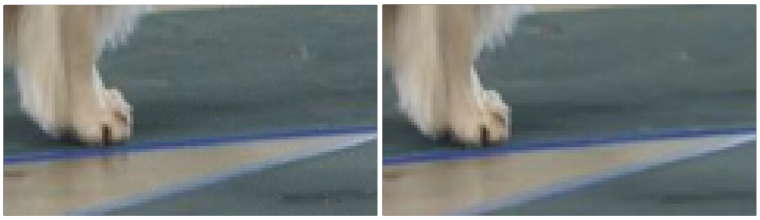
where γ is a scalar within the range $0 < \gamma < 1$ that defines the step size, while $\bar{\alpha}_t$ represents the noise schedule from Equation (5.5). After the variance preservation step, a DDIM step is applied to further denoise and refine the latent:

$$y_{t-1} = \text{DDIMStep}(y_t^M)$$

This two-step process progressively guides the latent towards the high-probability region associated with clean images. The gradient term $\nabla \log p_t(y_t^{m-1})$ is efficiently computed using the pre-trained diffusion model, specifically by relating it to the predicted noise ϵ_θ:

$$\nabla \log p_t\left(y_t^{m-1}\right) = -\frac{\epsilon_\theta\left(y_t^{m-1}, t\right)}{\sqrt{1-\bar{\alpha}_t}}$$

By iteratively applying VPS, we remove the artifacts of the degraded image by leveraging the priors captured by the pre-trained diffusion model.



(a) (b)

Figure 5.5: Comparison of inpainted image artifacts and purified results ssing DreamClean. **(a)** Image after inpainting, with artifacts; **(b)** Purified image after DreamClean processing.

Since our primary goal is to eliminate the artifacts introduced by the masking, we opted for a fast inverse approach in our implementation. Instead of using the DDIM inverse, we utilized the method described in Equation (5.5).

Bridging the gap between theoretical formulation and practical implementation, Algorithm 7 presents the details of the removing artifacts algorithm, showing the mathematical principles described above.

Algorithm 7: Removing artifacts algorithm

Require: $y, \tau, M, \eta_l, \eta_g$, A pre-trained diffusion model ϵ_θ
Ensure: The purified image y_0
$y_0 \leftarrow y$ // DDIM inversion, producing the latent y_τ
$y_\tau = \sqrt{\bar{\alpha}_\tau} y_0 + \sqrt{1-\bar{\alpha}_\tau}\epsilon$
for $t = \tau$ *to* 1 **do**
  $y_0^t \leftarrow y_t$ // Variance Preservation Sampling, no change to t
  for $l = 0$ *to* $L-1$ **do**
    $y_{l+1}^t \leftarrow y_l^t - \eta_l \frac{\epsilon_\theta(y_l^t, t)}{\sqrt{1-\bar{\alpha}_t}} + \eta_g \epsilon$
  end
  $y_t \leftarrow y_L^t$ // DDIM Step, from t to $t-1$
  $y_{t-1} \leftarrow \sqrt{\bar{\alpha}_{t-1}} \left(\frac{y_t - \sqrt{1-\bar{\alpha}_{t-1}} \epsilon_\theta(y_t, t)}{\sqrt{\bar{\alpha}_t}} \right) + \sqrt{1-\bar{\alpha}_{t-1}} \epsilon_\theta(y_t, t)$
end
return y_0

In summary, we first carefully mask and inpaint the sensitive regions, then we use DreamClean with its VPS technique to polish the image and remove any residual artifacts introduced by the unmasking strategy.

5.5. EXPERIMENTS

We test effectiveness of MSID under several attack methods, including AutoAttack [34], Projected Gradient Descent (PGD) [8], spatially transformed attacks (StAdv) [60], cADV [35] and SPSA [61]. We measure our defense performance using standard and robust accuracy metrics. Section 5.5.1 describes our experimental setup and Section 5.5.2 compares MSID to existing state-of-the-art AT and AP methods.

5.5.1. EXPERIMENTAL SETUP

Datasets and network architectures: Our evaluation involves two datasets: CIFAR-10 (with ResNet and WideResNet architectures) and ImageNet (with ResNet and ViT [62, 63]). We benchmark MSID against state-of-the-art defenses from RobustBench [64] and compare with other AP methods using their published settings. For the purification process, we utilize pre-trained diffusion models based on a U-Net architecture [65] from Ho et al. [25] for CIFAR-10, while employing models from Dhariwal & Nichol [28] for ImageNet.

Adversarial attacks: The robustness of MSID is evaluated against a diverse set of adversarial attacks. We implement AutoAttack [34], a benchmark encompassing both white-box and black-box attacks (l_∞ and l_2 threat models). To assess resilience against non-l_p-norm attacks, we include spatially transformed adversarial examples (StAdv) [60]. Furthermore, we use the PGD to conduct additional experiments.

MSID, which relies on multiple passes through neural networks, might encounter challenges with obfuscated gradients [66]. To address this, we test our defense method against strong adaptive attacks, including the BPDA+EOT attack [66]. We evaluate the effectiveness of this attack against our defense on the CIFAR-10 dataset. Finally, we generate adversarial examples using the score-based black-box attack, SPSA [61] and a color-based adversarial attack cADV [35], to ensure a comprehensive evaluation.

Evaluation metrics: We assess the effectiveness of the defense mechanisms using two key metrics: standard accuracy (on clean data) and robust accuracy (on adversarially perturbed data). Given the considerable computational cost of evaluating models against a wide range of attacks, we adopted the approach of Nie et al. [19], using a random subset of 512 test images for our robust accuracy evaluation.

Implementation details: All experiments in this paper are conducted using the hyperparameters detailed below and implemented in PyTorch [67] on an NVIDIA V100 GPU. For multi-scale superpixel segmentation, we use three scales: 5, 10, and 15 superpixels for CIFAR-10 and 100, 200, and 300 superpixels for ImageNet. For each scale, we assign weights as follows: $w_1 = 0.2$, $w_2 = 0.3$, and $w_3 = 0.5$. DiffPIR relies on four key hyperparameters: T (timesteps), u (resampling iterations), λ (conditioning guidance strength), and ζ (noise level). DreamClean introduces three additional hyperparameters: τ (inverse strength), and γ (step size). Through empirical optimization, we determine the following optimal values: $T = 20$, $u = 10$, $\lambda = 10$, $\zeta = 0.5$, and $\gamma = 0.05$. The inverse strength parameter, τ, is set to 50 for CIFAR-10 and 100 for ImageNet. In the SLIC algorithm, we set the compactness parameter to 10, while all other parameters are left at their default values as specified in the scikit-image library [68]. Selecting the top-3 clusters based on average

sensitivity ensures that coherent image regions, rather than isolated superpixels, are masked.

Table 5.1: Standard and robust accuracy against AutoAttack on CIFAR-10 with WideResNet classifier models.

(a) AutoAttack l_∞ ($\epsilon = 8/255$)

	Type	Method	Extra Data	Standard Acc.	Robust Acc.
WideResNet-28-10	AT	Zhang et al. [69]	✓	85.36	59.96
		Wu et al. [70]	✓	88.25	60.04
		Gowal et al. [71]	✓	89.48	62.70
		Cui et al. [72]	×	**92.16**	67.73
	AP	Yoon et al. [21]	×	85.66	68.42
		Lee & Kim [73]	×	90.16	79.11
		Nie et al. [19]	×	89.02	78.21
		Lin et al. [30]	×	90.62	82.76
		Ours	×	90.23	**87.89**
WideResNet-70-16	AT	Rebuffi et al. [74]	×	88.54	64.25
		Gowal et al. [71]	✓	91.10	65.87
		Rebuffi et al. [74]	✓	92.23	66.58
		Wang et al. [75]	×	**93.25**	70.69
	AP	Yoon et al. [21]	×	86.76	60.86
		Nie et al. [19]	×	90.43	74.83
		Lee & Kim [73]	×	90.53	76.35
		Lin et al. [30]	×	91.99	80.12
		Ours	×	90.86	**85.54**

(b) AutoAttack l_2 ($\epsilon = 0.5$)

	Type	Method	Extra Data	Standard Acc.	Robust Acc.
WideResNet-28-10	AT	Augustin et al. [76]	✓	92.23	77.93
		Rebuffi et al. [74]	×	91.79	78.32
		Wang et al. [75]	×	**95.16**	83.68
	AP	Yoon et al. [21]	×	85.66	83.51
		Lin et al. [30]	×	90.62	85.77
		Nie et al. [19]	×	91.4	86.43
		Lee & Kim [73]	×	90.16	87.29
		Ours	×	90.23	**88.28**
WideResNet-70-16	AT	Gowal et al. [71]	×	90.90	74.03
		Rebuffi et al. [74]	✓	**95.74**	81.44
	AP	Nie et al. [19]	×	92.68	85.70
		Lin et al. [30]	×	91.99	86.78
		Lee & Kim [73]	×	90.53	87.39
		Ours	×	90.23	**88.86**

5.5.2. COMPARISON WITH THE STATE-OF-THE-ART

We test our new defense against AutoAttack, using both l_∞ and l_2 attack methods. The performance is compared with state-of-the-art adversarial training (AT) defenses from RobustBench and state-of-the-art adversarial purification (AP) defenses from the literature. As shown in Tables 5.1a, 5.1b and 5.4b, our defense achieves state-of-the-art on CIFAR-10 (using WideResNet-28-10 and WideResNet-70-16) and ImageNet (using ResNet-50, ViT and WideResNet-50-2). Importantly, we achieve this without using extra data, unlike some competing methods. Specifically, our method improves robust accuracy by 5.13% and 5.42% compared to the second-best defense against l_∞ attacks on WideResNet-28-10 and WideResNet-70-16 respectively, and by 8.57%, 11.18% and 10.75% on ResNet-50, WideResNet-50-2 and ViT for ImageNet (see Table 5.4b). Against l_2 attacks on CIFAR-10, we observe a 1.47% improvement over the best AP defense. These consistent improvements across different datasets and adversarial attacks strongly suggest our adversarial purification technique is a highly effective defense strategy.

5.5.3. DEFENSE AGAINST UNSEEN ATTACKS

AT suffers from poor generalization to unseen attacks. Models robust to one attack may be vulnerable to others. This is proved by evaluating methods against three attacks (AutoAttack l_∞, l_2 and StAdv) as shown in Table 5.2 (seen attacks are underlined). While standard AT methods (e.g., Adv Train with l_∞) perform poorly against unseen attacks (e.g., l_2 and StAdv), our method shows robustness across all three. Table 5.2 illustrates that AT methods are constrained in their ability to defend against novel attacks, as they are effective only against the specific adversarial examples they were trained on. Conversely, AP methods demonstrate strong generalization capabilities, effectively defending against previously unseen attacks. Compared to these AP defenses, our approach achieves substantially higher robust accuracy, with improvements of 45.5%, 50.67%, and 36.9% on l_∞, l_2, and StAdv respectively.

Table 5.2: Standard accuracy and robust accuracy against AutoAttack l_∞ ($\epsilon = 8/255$), l_2 ($\epsilon = 1$) and StAdv non-l_p ($\epsilon = 0.05$) threat models on CIFAR-10 with ResNet-50 model.

Method	**Standard Acc.**	**AA l_∞**	**AA l_2**	**StAdv**
Standard Training	94.8	0.0	0.0	0.0
AT with l_∞ [10]	86.8	<u>49.0</u>	19.2	4.8
AT with l_2 [10]	85.0	39.5	<u>47.8</u>	7.8
AT with StAdv [10]	86.2	0.1	0.2	<u>53.9</u>
AT with all [10]	84.0	<u>25.7</u>	<u>30.5</u>	<u>40.0</u>
PAT-self [10]	82.4	30.2	34.9	46.4
Adv. CRAIG [12]	83.2	40.0	33.9	49.6
DiffPure [19]	88.2	79.57	81.29	68.73
AToP [30]	89.1	83.42	82.44	70.59
Ours	88.7	**85.5**	**84.57**	**86.5**

5.5.4. ROBUST EVALUATION OF DIFFUSION-BASED PURIFICATION

We comprehensively evaluate the robustness of our proposed method using the PGD attack. Our method demonstrates substantially improved robustness, achieving 86.52% and 89.25% robust accuracy against l_∞ and l_2 attacks, respectively, on WideResNet-28-10, and 85.54% and 88.08% on WideResNet-70-16. This represents a 2.49% and 2.16% improvement over the state-of-the-art for WideResNet-28-10, and 1.62% and 2.04% for WideResNet-70-16. Furthermore, on ImageNet (see Table 5.4a), our method shows a 8.57% improvement in robust accuracy for ResNet-50, 11.18% for WideResNet-50-2 and 10.75% for ViT.

Table 5.3: Standard and robust accuracy against PGD on CIFAR-10 with WideResNet classifier models.

(a) PGD l_∞ ($\epsilon = 8/255$)

	Type	Method	Extra Data	Standard Acc.	Robust Acc.
WideResNet-28-10	AT	Pang et al. [77]	✓	88.62	64.95
		Gowal et al. [71]	✓	88.54	65.93
		Gowal et al. [78]	✓	87.51	66.01
	AP	Yoon et al. [21]	×	85.66	79.28
		Nie et al. [19]	×	91.41	80.78
		Wang et al. [22]	×	**93.50**	81.05
		Lee & Kim [73]	×	90.16	84.03
		Ours	×	90.23	**86.52**
WideResNet-70-16	AT	Gowal et al. [71]	✓	91.10	68.66
		Gowal et al. [78]	✓	88.75	69.03
		Rebuffi et al. [74]	×	**92.22**	69.97
	AP	Yoon et al. [21]	×	86.76	74.58
		Nie et al. [19]	×	92.15	79.15
		Wang et al. [22]	×	**93.50**	83.33
		Lee & Kim [73]	×	90.53	83.92
		Ours	×	90.86	**85.54**

(b) PGD l_2 ($\epsilon = 0.5$)

	Type	Method	Extra Data	Standard Acc.	Robust Acc.
WideResNet-28-10	AT	Pang et al. [79]	✓	90.93	83.75
		Rebuffi et al. [74]	×	91.79	85.05
		Augustin et al. [76]	✓	**93.96**	86.14
	AP	Yoon et al. [21]	×	85.66	80.13
		Nie et al. [19]	×	91.41	83.11
		Wang et al. [22]	×	**93.50**	85.76
		Lee & Kim [73]	×	90.16	87.09
		Ours	×	90.23	**89.25**
WideResNet-70-16	AT	Rebuffi et al. [74]	×	92.41	86.24
		Gowal et al. [71]	✓	94.74	88.18
		Rebuffi et al. [74]	×	**95.74**	89.62
	AP	Yoon et al. [21]	×	86.76	78.54
		Nie et al. [19]	×	92.15	82.93
		Wang et al. [22]	×	**93.50**	84.16
		Lee & Kim [73]	×	90.53	86.04
		Ours	×	90.86	**88.08**

Table 5.4: Comparison of standard accuracy and robust accuracy against PGD l_∞ ($\epsilon = 4/255$) and AutoAttack l_∞ ($\epsilon = 4/255$) on ImageNet.

(a) PGD l_∞

Type	Method	Standard Acc.	Robust Acc.
AT	Wong et al. [13]	53.83	28.04
	Salman et al.[80]	63.86	39.11
AP	Nie et al.[19]	71.48	50.59
	Lee & Kim [73]	70.74	54.73
	Wang et al. [22]	70.17	67.06
	Ours	**76.39**	**74.60**

(b) AutoAttack l_∞

	Type	Method	Standard Acc.	Robust Acc.
ResNet-50	AT	Wong et al. [13]	55.62	26.95
		Salman et al.[80]	64.02	37.89
		Bai et al.[81]	67.3	35.51
	AP	Nie et al.[19]	67.79	66.23
		Ours	**76.39**	**74.80**
WRN-50-2	AT	Salman et al.[80]	68.46	39.25
	AP	Nie et al.[19]	71.16	64.21
		Ours	**78.60**	**75.39**
ViT	AT	Bai et al.[81]	66.50	35.50
	AP	Nie et al.[19]	73.63	52.33
		Ours	**75.11**	**63.08**

Figure 5.6: Standard accuracy and robust accuracy against SPSA l_∞ ($\epsilon = 8/255$) on CIFAR-10 with WideRestNet-28-10.

Method	Standard Acc.	Robust Acc.
Yoon et al. [21]	86.14	80.80
Wang et al.[22]	93.50	87.44
Ours	90.23	86.52

5.5.5. DEFENSE AGAINST SCORE-BASED BLACK-BOX ATTACK

Even without direct access to a model or its gradients, attackers can effectively estimate gradients using a large number of samples. One such method, SPSA [61], approximates gradients through a finite-difference approach, sampling points near an input. This gradient estimation allows attackers to craft adversarial perturbations even when model internals are unavailable. To evaluate the robustness of our defense against such attacks, we test it against SPSA with an l_∞ constraint ($\epsilon = 8/255$) on CIFAR-10 using WideRestNet-28-10. As Figure 5.6 shows, our method achieves state-of-the-art performance, reaching 86.52% robust accuracya mere 0.92% below Wang's defense [22].

5.5.6. DEFENSE AGAINST COLOR-BASED ADVERSARIAL ATTACK

cADV [35] generates adversarial examples by manipulating the color of an image. It leverages a pre-trained colorization model, guiding it to produce colorizations that misclassify the image while maintaining a natural appearance. This differs from other attacks that typically add small, imperceptible perturbations constrained by L_p norms. cADV, however, introduces large, smooth, and semantically consistent color changes. Instead of minimizing L_p distance, cADV searches the color space for adversarial examples, exploiting the colorization model's inherent understanding of natural color relationships and boundaries. Our proposed method demonstrates notable performance against the cAdv attack on both, CIFAR-10 and Imagenet datasets, as shown in Table 5.7.

Figure 5.7: Robust Accuracy of MSID against cAdv attack.

Dataset	Model	Robust Acc.
CIFAR-10	Resnet-50	75.19
	WRN-28-10	74.15
	WRN-70-16	72.46
Imagenet	Resnet-50	64.84

5.5.7. DEFENSE AGAINST STRONG ADAPTIVE ATTACKS

It is important to note that for AP methods that involve optimization loops or non-differentiable operations, BPDA attack is widely recognized as the most effective attack [82]. When considering stochastic defense methods, BPDA+EOT attack has become the standard benchmark for evaluating the latest advancements in adversarial purification [20, 21, 73]. For our evaluation we used EOT with 20 iterations.

Figure 5.8: Standard and robust accuracy against BPDA+EOT on CIFAR-10 with WideResNet-28-10.

Method	Standard Acc.	Robust Acc.
BPDA 50+EOT Hill et al. [20]	84.12	54.9
BPDA 40+EOT Yoon et al. [21]	86.14	70.01
BPDA 40+EOT Wang et al. [22]	**93.50**	**79.83**
BPDA 40+EOT Ours	90.23	74.21

5.6. ABLATION STUDIES

IMPACT OF CONTEXTUAL CUES ON ROBUSTNESS TO ADVERSARIAL ATTACKS

To assess the impact of incorporating contextual cues in our inpainting method, we evaluate the robust accuracy of our model against a comprehensive set of adversarial attacks previously tested on CIFAR-10 and ImageNet datasets, excluding contextual cues. The results in Table 5.5 demonstrate a significant decrease in robust accuracy when the contextual cues are removed from the inpainting process.

Table 5.5: Impact of contextual cues on robustness.

Attack	Dataset	Attack parameters	Model	Robust Acc.
AutoAttack	CIFAR-10	$l_\infty(\epsilon = 8/255)$	WRN-28-10	46.67
	Imagenet	$l_\infty(\epsilon = 8/255)$	Resnet-50	57.96
SPSA	CIFAR-10	$l_\infty(\epsilon = 8/255)$	WRN-28-10	50.39
PGD	CIFAR-10	$l_\infty(\epsilon = 8/255)$	WRN-28-10	51.95
	Imagenet	$l_\infty(\epsilon = 8/255)$	Resnet-50	64.84
StAdv	CIFAR-10	non-$l_p(\epsilon = 0.05)$	Resnet-50	52.92

ANALYSING THE CONTRIBUTION OF MULTI-SCALE INFORMATION IN OCCLUSION SENSITIVITY MAPPING

The number of scales used in our defense directly influences its robust accuracy. We assess the impact of varying the number of scales (see Figure 5.9) under PGD l_∞ attacks with $\epsilon = 8/255$ on CIFAR-10. Robust accuracy increased sharply with each

additional scale up to three scales. However, beyond three scales, the gains in robustness plateaued. Therefore, to balance the computational overhead introduced by incorporating more scales against the resulting improvement in robust accuracy, we determine that three scales provide the optimal trade-off. While adding scales offers negligible robustness, they require a greater computational cost, making three scales the most efficient and effective choice for our defense. This multi-scale approach achieves greater robustness than a single-scale alternative by capturing both finer image details and adversarial perturbations.

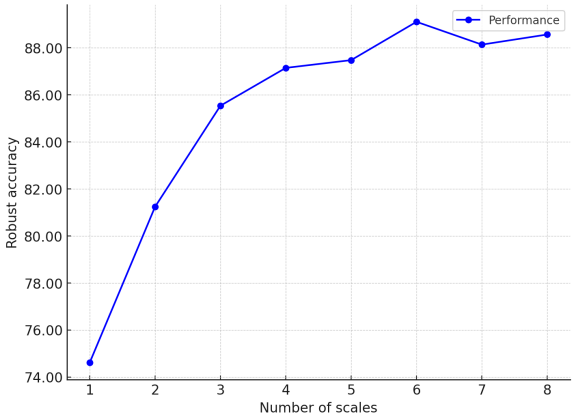


Figure 5.9: Impact of single vs multi-scale sensitivity maps on robust accuracy.

EVALUATION OF BINARY MASK GENERATION TECHNIQUES

Identifying the optimal strategy for converting the continuous sensitivity map into a binary mask can effectively guide the inpainting process toward the removal of adversarial perturbations. Two primary mechanisms in MSID framework, thresholding and clustering, are evaluated across various hyperparameter setups to analyze their effectiveness against PGD attacks with $\epsilon = 8/255$.

Thresholding: We first evaluate the thresholding approach by sorting pixel sensitivity values in decreasing order and selecting the top 5%, 10%, 20%, and 30% as potentially perturbed regions. The robust accuracy of our defense is then measured for each of these thresholds. In Figure 5.10, it shows a linear increase in robust accuracy up to the 20% threshold, followed by a sharp decrease at 30%. This suggests that while masking a small percentage of the most sensitive pixels effectively removes adversarial perturbations, masking too large a region negatively impacts performance. This decrease likely results from the removal of benign image features important for correct classification. By masking beyond the 20% threshold, the inpainting process not only removes potential adversarial perturbations, but also eliminates essential contextual information, reducing the classifier's ability to make accurate predictions.

Clustering: Subsequently, we investigate a clustering-based approach using k-means with k values of 3, 5, and 7. Similar to the thresholding approach, we selecte the

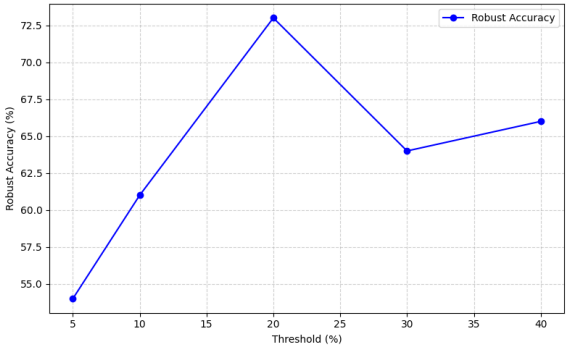


Figure 5.10: Robust accuracy vs. threshold percentage for mask generation using sensitivity-based thresholding. The plot shows that performance peaks at 20% and then decreases, suggesting that masking too many pixels removes important benign features crucial for accurate classification.

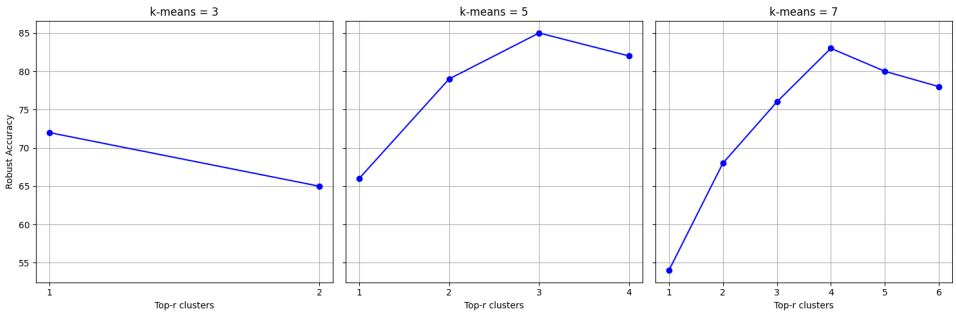


Figure 5.11: Robust accuracy vs. number of top-r clusters selected for inpainting using k-means clustering. Results are shown for $k = 3$ (left), $k = 5$ (middle), and $k = 7$ (right). Each plot demonstrates a peak in performance followed by a slight decrease, indicating the trade-off between removing adversarial perturbations and preserving benign features.

top-r clusters based on their centroid sensitivity and calculated the robust accuracy for each configuration. Figure 5.11 presents the performance of these experiments. Each k value exhibits a peak in robust accuracy followed by a slight decrease as more clusters are selected for inpainting. This trend mirrors the observations from the thresholding approach, reinforcing the balance between removing adversarial perturbations and preserving benign features. Selecting too many clusters for inpainting results in the loss of crucial image information, ultimately diminishing the defense's effectiveness.

When comparing the two approaches, k-means clustering consistently yields better robust accuracy than thresholding. This superior performance can be attributed to

the clustering algorithm's ability to capture the inherent structure of important image regions. By grouping pixels based on both sensitivity and spatial proximity, clustering potentially identifies perturbed regions that a rank-based thresholding approach might miss. Furthermore, clustering ensures that connected regions, rather than isolated superpixels, are targeted for inpainting, preserving contextual information and minimizing the disruption of benign features. This characteristic proves essential in maintaining the integrity of the image and facilitating accurate classification even after inpainting. Therefore, we conclude that clustering-based mask generation offers a more robust and effective approach for adversarial purification in MSID compared to simple thresholding techniques.

EVALUATING IMPUTATION STRATEGIES FOR ROBUST OCCLUSION SENSITIVITY MAPS

We investigate the impact of different imputation strategies on the robustness of MSID against adversarial attacks, specifically focusing on the generation of the occlusion sensitivity map. This map identifies important image regions for classification. We hypothesize that the method used to fill occluded regions during sensitivity analysis would significantly influence the robustness of the defense. Three imputation strategies were evaluated: (1) Zero-value imputation; (2) Histogram-based imputation and (3) Blurring-based imputation. These represent a set of approaches, from simple constant filling to more context-aware methods. Each imputation method is tested on images perturbed by a PGD attack with $\epsilon = 8/255$.

Our results demonstrate a clear difference in the robustness achieved with different imputation methods. As shown in Figure 5.12, zero-value imputation results in a robust accuracy of 79%, while histogram-based imputation achieves 76%. Blurring-based imputation outperforms the others, achieving a robust accuracy of 86%.

Blurring-based imputation's superior performance could arise from its ability to preserve contextual information. The Gaussian blur, unlike the sharp changes introduced by zero-value or even the potentially harsh constant color fills of histogram-based imputation, hides features rather than erasing them. This minimizes the introduction of out-of-distribution (OOD) artifacts by maintaining contextual similarity in occluded areas. By smoothing out sharp transitions and preserving the overall image structure, blurring can bring adversarial examples closer to their ground-truth labels, potentially reversing adversarial properties.

In contrast, zero-value imputation, while simple, creates discontinuities in the image. These artificial boundaries introduce unnatural features that are not present in the original data distribution. Furthermore, the complete removal of information in the occluded region can lead to a loss of crucial contextual cues that might be necessary for accurate sensitivity maps generation.

Histogram-based imputation, while attempting to maintain some color consistency, suffers from a similar, although less extreme issue. While the filled regions are less harsh than zero-value imputation, the constant color blocks can still introduce unnatural patterns. Additionally, the random sampling from the histogram, while preserving some color characteristics, fails to capture the spatial structure and texture of the occluded region, further obstructing accurate sensitivity map generation.

Therefore, our findings suggest that preserving contextual information during occlusion is essential for generating a robust sensitivity map and consequently, for enhancing the robustness of MSID against adversarial attacks. Blurring-based imputation, by maintaining this contextual similarity and minimizing OOD artifacts, provides the most effective strategy for occlusion in this context.

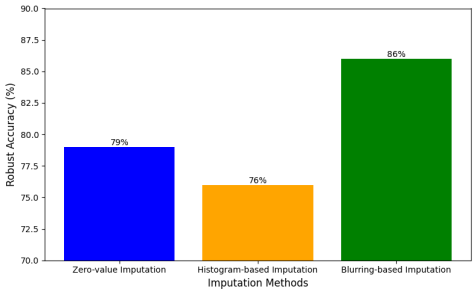


Figure 5.12: Robust accuracy of MSID under PGD attack (epsilon=8/255) using different imputation strategies during occlusion sensitivity map generation. Blurring-based imputation demonstrates superior robustness (86%) compared to zero-value (79%) and histogram-based (76%) imputation.

5.7. CONCLUSION

This work proposes MSID, an AP-based defense that leverages DDPM and occlusion sensitivity maps. MSID pioneers the use of DDPM inpainting for targeted perturbation removal, neutralizing adversarial perturbations while preserving benign features. Multi-scale occlusion analysis guides the inpainting process for precise attack mitigation. Extensive experiments demonstrate MSID's state-of-the-art performance and superior generalization to unseen attacks, including effectiveness against color-based attacks beyond traditional L_p-norm perturbations. This underscores the potential of integratin XAI and generative models for robust defense.

5.8. LIMITATIONS AND FUTURE WORK

5.8.1. LIMITATIONS

This work presents a novel approach to AP using multi-scale inpainting with DDPMs. Despite the promising results, several limitations should be acknowledged:

Computational cost: MSID relies on multi-scale superpixel segmentation, occlusion sensitivity map generation, and iterative DDPM inpainting. These processes, especially the DDPM component, are computationally expensive, potentially limiting the applicability of MSID in real-time or resource-constrained environments.

Dependence on pre-trained DDPMs: The effectiveness of MSID is tied to the quality of the pre-trained DDPM. Performance might vary depending on the dataset and architecture the DDPM was trained on, potentially requiring retraining or fine-tuning

for optimal performance across different tasks. Furthermore, the availability of pre-trained DDPMs for specific datasets or domains can be a limitation.

Limited evaluation against adaptive attacks: While the current evaluation includes a gray-box scenario (as detailed in Section 5.4) and testing against BPDA+EOT, a more comprehensive assessment against a wider array of adaptive attacks is important. The specific mechanisms of MSID, particularly the occlusion sensitivity maps, could potentially be exploited by adaptive adversaries to create stronger adversarial examples.

Hyperparameter Sensitivity: MSID introduces several hyperparameters related to superpixel segmentation, occlusion sensitivity analysis, DDPM inpainting, and artifact removal. The optimal values for these hyperparameters vary across datasets and attack types, requiring careful tuning. A more thorough investigation of hyperparameter sensitivity and potential automated tuning methods could further enhance the defense.

Limited comparison against diverse defense strategies: This work benchmarks MSID primarily against DM-based adversarial purification and AT. While demonstrating state-of-the-art performance within this subset of defenses, a broader evaluation covering other defense strategies, including certified defenses or non-diffusion based AP is necessary for a more comprehensive assessment of MSID's capabilities.

5.8.2. FUTURE WORK

This work opens up some interesting research directions. One key area is boosting the speed of our method. Right now, the DDPM inpainting step is a computational bottleneck. We need to explore faster ways to sample from DDPMs or even entirely different inpainting methods that are less computationally intensive without compromising image quality. Improving how we approximate the occlusion sensitivity maps could also help speed things up. Furthermore, investigating alternative Explainable AI (XAI) techniques for guiding the inpainting process could lead to more efficient and targeted restorations.

We also need to make MSID more robust against adaptive attacks. Our current testing has not covered the full spectrum of adversarial attacks, particularly strong adaptive attacks

Setting the right hyperparameters for MSID can be tricky. Automating this process, perhaps using would make the method much easier to use and more adaptable to different datasets and tasks.

Finally, we need a more comprehensive comparison of MSID against other defense strategies. This includes certified defenses and purification methods that don't rely on diffusion models. A more comprehensive comparison will help us understand MSID's strengths and weaknesses compared to other defenses.

REFERENCES

[1] A. Kurakin, I. J. Goodfellow and S. Bengio. 'Adversarial Machine Learning at Scale'. In: *5th International Conference on Learning Representations* (2016).

[2] S. M. Moosavi-Dezfooli, A. Fawzi and P. Frossard. 'DeepFool: a simple and accurate method to fool deep neural networks'. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2015), pp. 2574–2582.

[3] J. Su, D. V. Vargas and S. Kouichi. 'One pixel attack for fooling deep neural networks'. In: *IEEE Transactions on Evolutionary Computation* 23.5 (2017), pp. 828–841.

[4] M. Andriushchenko, F. Croce, N. Flammarion and M. Hein. 'Square Attack: a query-efficient black-box adversarial attack via random search'. In: *Lecture Notes in Computer Science* 12368 LNCS (2019), pp. 484–501.

[5] I. J. Goodfellow, J. Shlens and C. Szegedy. 'Explaining and Harnessing Adversarial Examples'. In: (2014).

[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow and R. Fergus. 'Intriguing properties of neural networks'. In: (2013).

[7] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui and M. I. Jordan. 'Theoretically Principled Trade-off between Robustness and Accuracy'. In: (2019).

[8] A. Madry, A. Makelov, L. Schmidt, D. Tsipras and A. Vladu. 'Towards Deep Learning Models Resistant to Adversarial Attacks'. In: (2017).

[9] O. Poursaeed, T. Jiang, H. Yang, S. Belongie and S.-N. Lim. *Robustness and Generalization via Generative Adversarial Training*. Tech. rep.

[10] C. Laidlaw, S. Singla and S. Feizi. 'Perceptual Adversarial Robustness: Defense Against Unseen Threat Models'. In: (2020).

[11] J. Tack, S. Yu, J. Jeong, M. Kim, S. J. Hwang and J. Shin. 'Consistency Regularization for Adversarial Robustness'. In: *AAAI Conference on Artificial Intelligence* 36 (2021), pp. 8414–8422.

[12] H. M. Dolatabadi, S. Erfani and C. Leckie. 'Robustness and Beyond: Unleashing Efficient Adversarial Training'. In: *Conference on Learning Representations*, (2021).

[13] E. Wong, L. Rice and J. Z. Kolter. 'Fast is better than free: Revisiting adversarial training'. In: (2020).

[14] V. Srinivasan, A. Marban, K.-R. Müller, W. Samek and S. Nakajima. 'Robustifying Models Against Adversarial Attacks by Langevin Dynamics'. In: (2018).

[15] C. Shi, C. Holtz and G. Mishne. ‘Online Adversarial Purification based on Self-Supervision’. In: (2021).

[16] Y. Yang, G. Zhang, D. Katabi and Z. Xu. ‘ME-Net: Towards Effective Adversarial Robustness with Matrix Estimation’. In: (2019).

[17] L. Schott, J. Rauber, M. Bethge and W. Brendel. ‘Towards the first adversarially robust neural network model on MNIST’. In: (2018).

[18] P. Ghosh, A. Losalka and M. J. Black. ‘Resisting Adversarial Attacks using Gaussian Mixture Variational Autoencoders’. In: (2018).

[19] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat and A. Anandkumar. ‘Diffusion Models for Adversarial Purification’. In: (2022).

[20] M. Hill, J. Mitchell and S.-C. Zhu. ‘Stochastic Security: Adversarial Defense Using Long-Run Dynamics of Energy-Based Models’. In: (2020).

[21] J. Yoon, S. J. Hwang and J. Lee. ‘Adversarial purification with Score-based generative models’. In: *Proceedings of Machine Learning Research* 139 (2021), pp. 12062–12072.

[22] J. Wang, Z. Lyu, D. Lin, B. Dai and H. Fu. ‘Guided Diffusion Model for Adversarial Purification’. In: (2022).

[23] F. Tramèr, N. Carlini and W. Brendel. *On Adaptive Attacks to Adversarial Example Defenses*. Tech. rep.

[24] S. Chen, Z. Huang, Q. Tao, Y. Wu, C. Xie and X. Huang. *Adversarial Attack on Attackers: Post-Process to Mitigate Black-Box Score-Based Query Attacks*. Tech. rep.

[25] J. Ho, A. Jain and P. Abbeel. ‘Denoising Diffusion Probabilistic Models’. In: (2020).

[26] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon and B. Poole. ‘Score-Based Generative Modeling through Stochastic Differential Equations’. In: (2020).

[27] A. Vahdat, K. Kreis and J. Kautz. *Score-based Generative Modeling in Latent Space*. Tech. rep.

[28] P. Dhariwal and A. Nichol. *Diffusion Models Beat GANs on Image Synthesis*. Tech. rep.

[29] Z. He, A. S. Rakin and D. Fan. *Parametric Noise Injection: Trainable Randomness to Improve Deep Neural Network Robustness against Adversarial Attack*. Tech. rep.

[30] G. Lin, C. Li, J. Zhang, T. Tanaka and Q. Zhao. ‘Adversarial Training on Purification (AToP): Advancing Both Robustness and Generalization’. In: (2024).

[31] D. Stutz, M. Hein and B. Schiele. ‘Confidence-Calibrated Adversarial Training: Generalizing to Unseen Attacks’. In: *International Conference on Machine Learning* (2019), pp. 9092–9103.

[32] M. D. Zeiler and R. Fergus. ‘Visualizing and Understanding Convolutional Networks’. In: (2013).

[33] J. Xiao, R. Feng, H. Zhang, Z. Liu, Z. Yang, Y. Zhu, X. Fu, K. Zhu, Y. Liu and Z.-J. Zha. *DREAMCLEAN: RESTORING CLEAN IMAGE USING DEEP DIFFUSION PRIOR*. Tech. rep.

[34] F. Croce and M. Hein. *Reliable Evaluation of Adversarial Robustness with an Ensemble of Diverse Parameter-free Attacks*. Tech. rep. 2020.

[35] A. Bhattad, M. J. Chong, K. Liang, B. Li and D. A. Forsyth. ‘Unrestricted Adversarial Examples via Semantic Manipulation’. In: *International Conference on Learning Representations* (2019).

[36] P. Samangouei, M. Kabkab and R. Chellappa. ‘Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models’. In: (2018).

[37] Y. Song, T. Kim, S. Nowozin, S. Ermon and N. Kushman. ‘PixelDefend: Leveraging Generative Models to Understand and Defend against Adversarial Examples’. In: (2017).

[38] Y. Du, I. Mordatch and G. Brain. *Implicit Generation and Modeling with Energy-Based Models*. Tech. rep.

[39] W. Grathwohl, K.-C. Wang, J.-H. Jacobsen, D. Duvenaud, M. Norouzi and K. Swersky. ‘Your Classifier is Secretly an Energy Based Model and You Should Treat it Like One’. In: (2019).

[40] G. Lin, Z. Tao, J. Zhang, T. Tanaka and Q. Zhao. *Adversarial Guided Diffusion Models for Adversarial Purification*.

[41] H. Chen, Y. Dong, Z. Wang, X. Yang, C. Duan, H. Su and J. Zhu. ‘Robust Classification via a Single Diffusion Model’. In: *Proceedings of Machine Learning Research* 235 (2023), pp. 6643–6665.

[42] F. Croce, S. Gowal, T. Brunner, E. Shelhamer, M. Hein and T. Cemgil. ‘Evaluating the Adversarial Robustness of Adaptive Test-time Defenses’. In: *Machine Learning Research* (2022), pp. 4421–4435.

[43] P. Dhariwal and A. Nichol. ‘Diffusion Models Beat GANs on Image Synthesis’. In: *Advances in Neural Information Processing Systems* 11 (2021), pp. 8780–8794.

[44] J. Song, C. Meng and S. Ermon. ‘Denoising Diffusion Implicit Models’. In: *International Conference on Learning Representations* (2020).

[45] S. Lapuschkin, S. Wäldchen, A. Binder, G. Montavon, W. Samek and K.-R. Müller. ‘Unmasking Clever Hans Predictors and Assessing What Machines Really Learn’. In: *Nature Communications* 10.1 (2019).

[46] S. Letzgus, P. Wagner, J. Lederer, W. Samek, K.-R. Müller and G. Montavon. ‘Toward Explainable AI for Regression Models’. In: *IEEE Signal Processing Magazine* 39.4 (2021), pp. 40–58.

[47] A. Ghorbani, J. Wexler, J. Zou and B. Kim. 'Towards Automatic Concept-based Explanations'. In: *Advances in Neural Information Processing Systems* 32 (2019).

[48] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas and R. Sayres. 'Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV)'. In: *International Conference on Machine Learning* 6 (2017), pp. 4186–4195.

[49] A. Goldstein, A. Kapelner, J. Bleich and E. Pitkin. 'Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation'. In: *Journal of Computational and Graphical Statistics* 24.1 (2013), pp. 44–65.

[50] D. W. Apley and J. Zhu. 'Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models'. In: *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 82.4 (2016), pp. 1059–1086.

[51] S. Blücher, J. Vielhaben and N. Strodthoff. 'Decoupling Pixel Flipping and Occlusion Strategy for Consistent XAI Benchmarks'. In: *Transactions on Machine Learning Research* (2024).

[52] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua and S. Süsstrunk. 'SLIC superpixels compared to state-of-the-art superpixel methods'. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.11 (2012), pp. 2274–2281.

[53] R. Fong and A. Vedaldi. 'Interpretable Explanations of Black Boxes by Meaningful Perturbation'. In: *IEEE International Conference on Computer Vision* (2017), pp. 3449–3457.

[54] X. Bi, Z. Yang, B. Liu, X. Cun, C.-M. Pun, P. Lio and B. Xiao. 'ZeroPur: Succinct Training-Free Adversarial Purification'. In: (2024).

[55] J. A. Hartigan and M. A. Wong. 'Algorithm AS 136: A K-Means Clustering Algorithm'. In: *Applied Statistics* 28 (1979), p. 100.

[56] Y. Zhu, K. Zhang, J. Liang, J. Cao, B. Wen, R. Timofte and L. Van Gool. 'Denoising Diffusion Models for Plug-and-Play Image Restoration'. In: (2023).

[57] D. Geman and C. Yang. 'Nonlinear Image Recovery with Half-Quadratic Regularization'. In: *IEEE Transactions on Image Processing* 4.7 (1995), pp. 932–946.

[58] B. Kawar, M. Elad, S. Ermon and J. Song. 'Denoising Diffusion Restoration Models'. In: *Advances in Neural Information Processing Systems* 35 (2022).

[59] H. Chung, J. Kim, M. T. McCann, M. L. Klasky and J. C. Ye. 'Diffusion Posterior Sampling for General Noisy Inverse Problems'. In: *International Conference on Learning Representations* (2022).

[60] C. Xiao, J.-Y. Zhu, B. Li, W. He, M. Liu and D. Song. 'Spatially Transformed Adversarial Examples'. In: (2018).

[61] J. Uesato, B. O'Donoghue, A. v. d. Oord and P. Kohli. 'Adversarial Risk and the Dangers of Evaluating Against Weak Attacks'. In: (2018).

[62] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit and N. Houlsby. 'An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale'. In: *International Conference on Learning Representations* (2020).

[63] T. Darcet, M. Oquab, J. Mairal and P. Bojanowski. 'Vision Transformers Need Registers'. In: *International Conference on Learning Representations* (2023).

[64] F. Croce, M. Andriushchenko, V. Sehwag, E. Debenedetti, N. Flammarion, M. Chiang and M. Hein. 'RobustBench: a standardized adversarial robustness benchmark'. In: *Neural Information Processing Systems* (2020).

[65] O. Ronneberger, P. Fischer and T. Brox. 'U-Net: Convolutional Networks for Biomedical Image Segmentation'. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9351 (2015), pp. 234–241.

[66] A. Athalye, N. Carlini and D. Wagner. 'Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples'. In: *International Conference on Machine Learning* 1 (2018), pp. 436–448.

[67] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala. 'PyTorch: An Imperative Style, High-Performance Deep Learning Library'. In: *Advances in Neural Information Processing Systems* 32 (2019).

[68] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu and t. s.-i. contributors the scikit-image. 'scikit-image: Image processing in Python'. In: *PeerJ* 2014.1 (2014).

[69] J. Zhang, J. Zhu, G. Niu, B. Han, M. Sugiyama and M. Kankanhalli. 'Geometry-aware Instance-reweighted Adversarial Training'. In: (2020).

[70] D. Wu, S.-t. Xia and Y. Wang. 'Adversarial Weight Perturbation Helps Robust Generalization'. In: (2020).

[71] S. Gowal, C. Qin, J. Uesato, T. Mann and P. Kohli. 'Uncovering the Limits of Adversarial Training against Norm-Bounded Adversarial Examples'. In: (2020).

[72] J. Cui, Z. Tian, Z. Zhong, X. Qi, B. Yu and H. Zhang. 'Decoupled Kullback-Leibler Divergence Loss'. In: (2023).

[73] M. Lee and D. Kim. 'Robust Evaluation of Diffusion-Based Adversarial Purification'. In: *IEEE International Conference on Computer Vision* (2023), pp. 134–144.

[74] S.-A. Rebuffi, S. Gowal, D. A. Calian, F. Stimberg, O. Wiles and T. Mann. 'Fixing Data Augmentation to Improve Adversarial Robustness'. In: (2021).

[75] Z. Wang, T. Pang, C. Du, M. Lin, W. Liu and S. Yan. 'Better Diffusion Models Further Improve Adversarial Training'. In: *Machine Learning Research* 202 (2023), pp. 36246–36263.

[76] M. Augustin, A. Meinke and M. Hein. 'Adversarial Robustness on In- and Out-Distribution Improves Explainability'. In: *Lecture Notes in Computer Science* 12371 LNCS (2020), pp. 228–245.

[77] T. Pang, M. Lin, X. Yang, J. Zhu and S. Yan. 'Robustness and Accuracy Could Be Reconcilable by (Proper) Definition'. In: *Machine Learning Research* (2022).

[78] S. Gowal, S. A. Rebuffi, O. Wiles, F. Stimberg, D. Calian and T. Mann. 'Improving Robustness using Generated Data'. In: *Advances in Neural Information Processing Systems* 6 (2021), pp. 4218–4233.

[79] V. Sehwag, S. Mahloujifar, T. Handina, S. Dai, C. Xiang, M. Chiang and P. Mittal. 'Robust Learning Meets Generative Models: Can Proxy Distributions Improve Adversarial Robustness?' In: *International Conference on Learning Representations* (2021).

[80] H. Salman, A. Ilyas, L. Engstrom, A. Kapoor and A. Madry. 'Do Adversarially Robust ImageNet Models Transfer Better?' In: *Advances in Neural Information Processing Systems* (2020).

[81] Y. Bai, J. Mei, A. Yuille and C. Xie. 'Are Transformers More Robust Than CNNs?' In: *Advances in Neural Information Processing Systems* 32 (2021), pp. 26831–26843.

[82] F. Tramèr, N. Carlini, W. Brendel and A. Madry. 'On Adaptive Attacks to Adversarial Example Defenses'. In: *Advances in Neural Information Processing Systems* 2020-December (2020).

6

DISCUSSION

DNNs have achieved remarkable progress across a wide range of real-world applications. Furthermore, model architectures such as CNNs and ViTs, coupled with diverse training paradigms, have proven highly effective in solving numerous computer vision tasks.

Recent studies have highlighted the susceptibility of DNNs to various security threats, including adversarial and backdoor attacks. Adversarial attacks undermine model robustness by exploiting small and imperceptible perturbations in input data, thereby causing incorrect outputs. Backdoor attacks, on the other hand, compromise data integrity by altering a small subset of samples with triggers to embed malicious behaviors, often referred to as *Trojans*, within the victim model. These vulnerabilities pose significant risks to the trustworthiness and reliability of DNN models, particularly in safety-critical applications, where such attacks can lead to catastrophic consequences.

In the field of adversarial machine learning, we investigate three pivotal research challenges related to the trustworthiness and reliability of DNN models in both centralized and decentralized settings. These challenges include stealthiness and robustness in data and model aspects.

This chapter first summarizes the contributions of the thesis, showing how our proposed methods address the security challenges and research questions outlined in Sections 1.2 and 1.3. We then discuss the limitations of the proposed methods and potential future works.

6.1. CONTRIBUTION IN BACKDOOR ATTACKS

In this section, we present a summary of our contributions, as outlined in Chapters 2 to 4, which address three research questions regarding the stealthiness and robustness of backdoor attacks. First, we focus on the design of robust and dual-domain stealthy triggers to perform backdoor attacks under the black-box setting. Then, we analyze the anomalies in feature representations and model parameters, and propose stealthy backdoor attacks against robust FL aggregation rules.

6.1.1. STEALTHINESS AND ROBUSTNESS IN CENTRALIZED BACKDOOR ATTACKS

The first problem we investigate in Chapter 2 is:

Q1: *How can effectiveness, dual-domain stealthiness, and robustness be simultaneously achieved in backdoor attacks under practical settings?*

To answer the question, we first constrain the trigger location in the low-frequency region to achieve robustness against image preprocessing methods. Then, we minimize frequency perturbations of the trigger and optimize frequency bands to maximize attack effectiveness and stealthiness. Since we cannot access the victim models in the practical settings, we search the optimal trigger via simulated annealing to effectively optimize two properties of the trigger in discrete spectral space on the surrogate models.

While addressing this research question, we first observe that even slight trigger perturbations in the frequency domain can be detected and learned by visual neural networks. Therefore, using untrusted or harmful datasets from the Internet to train private models poses a significant backdoor threat. Second, our solution has an inherent limitation in stealthiness against advanced white-box backdoor defenses. This is because, in practical scenarios, the attacker cannot access the victim model and thus cannot ensure stealthiness in the latent space. Finally, since we only insert the trigger into the dataset without modifying the model parameters, such malicious data manipulations can be effectively mitigated by our solution of **Q3**.

6.1.2. STEALTHINESS IN DECENTRALIZED BACKDOOR ATTACKS

The second problem we address in Chapter 3 is the following:

Q2: *How can malicious updates be disguised as benign ones at the parameter level to bypass current detection strategies while still maintaining the effectiveness of the backdoor attack?*

To answer the question, we first introduce a frequency trigger function that generates imperceptible poisoned samples to achieve stealthiness in the input space. Next, we propose a task-sensitive neuron searcher to identify backdoor neurons during backdoor training. We further constrain the impact of those backdoor neurons and restrain parameter dissimilarity between malicious and benign parameters. Together, these constraints ensure that malicious updates remain indistinguishable from benign ones, effectively bypassing existing FL defenses.

While addressing this research question, we prioritize the stealthiness in our trigger design, as visible triggers introduce distinctive backdoor features that can be detected and mitigated by trigger inversion techniques. In this solution, we manipulate only the model parameters and do not fully leverage the local data with advanced triggers to enhance stealthiness against FL defenses. Although the defender does not have access to the attacker's local training data, it is strongly

recommended to design advanced trigger patterns instead of relying on simple patch-based triggers. For example, a universal trigger pattern presents a limitation in these solutions, as each poisoned sample would exhibit a unified backdoor feature during activation, making the backdoor behavior easier to detect. Additionally, our solution by penalizing the influence of backdoor neurons and constraining the distance between model parameters is relatively straightforward and does not fundamentally address the abnormal parameter effects introduced during backdoor training. Finally, it is important to note that in decentralized settings, the attacker has an advantage in terms of knowledge, as they inherently have access to benign data and models, unlike the threat model in **Q1**.

Another problem we investigate in Chapter 4 is the following:

Q3: *How to eliminate the anomalies introduced during backdoor training while making the trigger sufficiently stealthy for inference under FL settings?*

To answer the question, we utilize a trigger generator to produce stealthy sample-specific triggers to achieve stealthiness in the input space. To solve the abnormalities introduced by backdoor training, we minimize the discrepancy in hidden features between poisoned data and their benign counterparts while training the generator. This idea naturally reduce the abnormality of creating an extra routing for backdoor since the latent features make poisoned data “looks like" benign ones with target label. Additionally, to make our triggers adaptive to the changes in global model, the generator is continuously trained across FL iterations. Finally, we formulate the process of finding optimal trigger generator and training malicious model in a bi-level, non-convex and constrained optimization problem, and achieve optimum by proposing a simple but practical optimization process.

While addressing this research question, we build upon the challenges discussed in **Q2** to further improve attack stealthiness against FL detections. We first enhance stealthiness in the input space by constraining the L_2 norm of the triggers generated by our trigger generator. Next, we introduce an approach to achieve stealthiness from a data perspective similar to attacks in centralized settings. As our sample-specific triggers generate feature representations closely resembling benign data, they fundamentally eliminate the anomalies typically introduced during backdoor training. Moreover, since the triggers are sample-specific, it is more difficult to leverage trigger inversion mechanisms to reverse flexible triggers compared to universal triggers. Our solution exposes a critical challenge to current FL defenses and motivates the development of more robust defensive techniques capable of detecting malicious clients from both data and model aspects.

6.2. CONTRIBUTION IN ADVERSARIAL DEFENSES

In this section, we present a summary of our contributions in Chapter 5, which address the research question related to the robustness of adversarial purification. We aim to the development of a targeted strategy for the localization and removal of adversarial perturbations, as well as for the restoration of images with the capability

of generative models.

6.2.1. ROBUSTNESS IN ADVERSARIAL PURIFICATION

The final problem we address in Chapter 5 is the following:

Q4: *How can adversarial perturbations be effectively mitigated while preserving fine-grained clean features to maintain high clean accuracy during adversarial purification?*

To answer the question, we first utilize multi-scale occlusion sensitivity maps to identify the target regions to be purified. This enables targeted purification, focusing on removing adversarial noise only from specific areas while preserving clean image features to maintain high clean accuracy. Unlike conventional diffusion model processes that disrupt the entire image, we leverage the DDPM inpainting technique to restore localized information, which precisely mitigates adversarial perturbations. To further achieve high clean accuracy, we utilize jittered grid unmasking for DDPM inpainting to keep critical contextual information required for accurate reconstruction.

Although our approach is primarily designed to defend against adversarial attacks, it can also be applied to purify backdoor triggers, particularly in black-box attacks that only manipulate input images, such as the method proposed in **Q1**. This effectiveness stems from the ability of our method to identify the most sensitive regions (trigger locations) for model prediction and to disrupt those pixels, thereby neutralizing the backdoor triggers. However, our solution may face challenges when addressing sparse adversarial or backdoor attacks, where adversarial perturbations or triggers compromise only a few scattered pixels. This limitation arises because our XAI tools are capable of identifying target regions rather than isolated single pixels. In the future, we plan to enhance our approach by replacing the current diffusion models with more advanced architectures – such as diffusion transformers for image inpainting – that leverage transformers as their backbone.

6.3. LIMITATIONS AND FUTURE WORKS

STEALTHINESS AND ROBUSTNESS IN CENTRALIZED BACKDOOR ATTACKS

Although we have made a significant advancement in dual-domain stealthy and robust backdoor attacks in Chapter 2, certain limitations remain in our approach including task diversity, optimization efficiency and attack effectiveness against state-of-the-art backdoor defense mechanisms.

Tasks. In this work, we concentrate on various computer vision tasks, which have been the focus of numerous existing works [1–5]. Extending our attack to other input domains (e.g., text) is challenging because applying DCT-based perturbations (triggers) to text embeddings can result in semantically incoherent output. However, our method can be easily integrated into backdoor attack frameworks against multimodal models, such as vision-language models [6, 7], which also rely on embedding triggers in the image domain. In the future, we intend to expand the scope of this work to other vision tasks, e.g., objection detection and semantic

segmentation. Our method (OD), prior works [8–10] use trigger injection function similar to Equation (1.8) (in Chapter 1) in classification tasks, enabling the insertion of frequency triggers into images. However, most mainstream attacks in OD rely on visible patch-based triggers to ensure successful capture by cameras in real-world applications, such as autonomous driving. Due to this dependency, our invisible frequency attack cannot be applied to physical OD tasks. Additionally, we plan to investigate the vulnerabilities of diffusion models against our proposed attack. Transferring our attack to diffusion models (DMs) is straightforward, as backdooring DMs [11, 12] relies on trigger injection functions similar to those used in classification tasks.

Optimization efficiency. The trigger search process is executed in a hybrid GPU-CPU environment during trigger evaluation and optimization phases. For example, searching optimal triggers for large-scale datasets during the optimization process can require several minutes or hours due to the computational complexity involved. It deserves further efforts to design a GPU-accelerated SA to minimize data transmission across hardware, thus improving the efficiency of our proposed attack.

Effectiveness against state-of-the-art backdoor defenses. Note that black-box attacks fail to achieve the same level of robustness against state-of-the-art backdoor defenses as white-box methods due to the lack of control over the training process of the victim model. This is further empirically supported by [13], which also proposes a black-box frequency imperceptible attack. The results indicate that while current black-box frequency attacks exhibit superior stealthiness in both spatial and frequency domains, they lack robustness against advanced white-box backdoor defenses. Unlike white-box attacks, black-box attacks inherently face this limitation. To defend against black-box attacks, defenders can also leverage image purification methods as in [14] and our proposed technique in Chapter 5 to remove malicious triggers from poisoned inputs. In conclusion, black-box backdoor attacks are less harmful than white-box attacks under robust defenses. Additionally, purifying poisoned samples under black-box attacks is often easier than detecting hidden trojans in compromised models.

STEALTHINESS IN DECENTRALIZED BACKDOOR ATTACKS

In Chapters 3 and 4, we address the challenge of enhancing stealthiness in backdoor attacks under decentralized settings. However, both methods exhibit shared limitations that warrant further investigation. For example, both proposed attacks focus on stealthiness, i.e., bypassing robust FL defenses, rather than durability, i.e., maintaining backdoor effectiveness in the global model after the attacker stops uploading poisoned updates.

Tasks. In Chapters 3 and 4, we concentrate on various computer vision tasks, which have been the focus of numerous existing works [15–20]. In the future, we intend to expand the scope of this work by applying our design to other real-world applications, such as natural language processing (NLP) and reinforcement learning (RL), as well as other vision tasks, e.g., object detection. As demonstrated in the limitations of centralized backdoor attacks, methods in Chapters 3 and 4 can be extended to other vision tasks, such as object detection. Since the triggers used in

Chapters 3 and 4 are imperceptible, slight modifications are necessary to ensure successful attacks in physical OD tasks. In Chapter 3, our frequency trigger can be replaced with a patch-based trigger. In Chapter 4, the natural stealthiness constraint in Equation (4.4) can be relaxed to allow visible triggers. The method in Chapter 3 can be easily extended to NLP tasks, as it only involves identifying malicious neurons and manipulating model parameters without requiring input modifications. However, the method in Chapter 4 requires an image generator to produce triggers, making it inapplicable in the text domain.

Computational cost. In Chapter 3, the step-forward training needs additional computational costs. To mitigate these expenses, we may consider enabling malicious agents to collaborate in generating a shared malicious update through split learning. Since malicious clients can collude in FL, generating a shared update is relatively straightforward. Similarly, in Chapter 4, the use of generative adversarial networks incurs additional training overhead. To address this limitation, future research could focus on developing more efficient and lightweight generative networks without the loss of attack performance. Additionally, it is feasible to leverage few-shot learning to reduce the number of training samples required for the generator, lowering its computation cost.

Durability. In Chapters 3 and 4, the primary focus of the methods is to achieve stealthiness rather than durability, in contrast to durable backdoor attacks. We emphasize that the durability of backdoor attacks on FL is orthogonal to stealthiness, and we leave it as an open problem. State-of-the-art methods, such as [21], propose a solution that ensures both attack stealthiness and durability. It achieves stealthiness by injecting the backdoor into specific layers, and enhances backdoor durability by avoiding frequently updated gradient coordinates. Such a philosophy can be applied to the method in Chapter 3 to enhance durability by avoiding updating certain parameters during the model camouflage phase.

ROBUSTNESS IN ADVERSARIAL PURIFICATION

In Chapter 5, we present a novel AP-based purification method using multi-scale inpainting with DDPMs to enhance the robustness of DNN models. Despite the promising results, several limitations should be acknowledged and solved in the future.

Computational cost. Our defense mechanism integrates multi-scale superpixel segmentation, occlusion sensitivity map generation, and iterative DDPM inpainting. While effective, these processes, particularly DDPM process, are computationally expensive, which may restrict the applicability of our approach in real-time or resource-constrained scenarios. Future research should prioritize the development of more efficient sampling methods for DDPMs or explore alternative inpainting strategies that balance computational efficiency with high-quality image restoration. It is feasible to accelerate our original pretrained DDPMs sampling process using a distribution-based distillation strategy. Specifically, [22] introduces the Denoising Student framework, which reduces iterative denoising steps through knowledge distillation. By applying this technique to our pretrained models, we can distill a student model that significantly speeds up sampling. Additionally, improving the

approximation of occlusion sensitivity maps could further enhance processing speed. Investigating alternative Explainable AI techniques to guide the inpainting process could lead to more efficient and targeted image restoration techniques. For instance, occlusion maps can be replaced with other XAI techniques, such as LIME [23], to assess the efficiency and robustness of our approach. Such a comparison would provide deeper insights into how XAI tools influence the image restoration process.

Hyperparameter sensitivity. Our proposed method incorporates several hyperparameters regarding superpixel segmentation, occlusion sensitivity analysis, DDPM inpainting, and artifact removal. The optimal configuration of these hyperparameters is highly dependent on the specific dataset and type of adversarial attacks, necessitating careful hyperparameter tuning. A detailed analysis of hyperparameter sensitivity, coupled with the development of automated tuning strategies, could significantly improve the adaptability of the defense mechanism across diverse datasets and tasks. However, running hyperparameter tuning algorithms, such as grid search, random search, and Bayesian optimization, is impractical in our framework, since evaluating performance for a single hyperparameter configuration already introduces high computational costs. It is possible to reduce hyperparameter sensitivity by avoiding using components that rely on large hyperparameters. For example, we can sacrifice image quality by avoiding DreamClean and its Variance Preservation Sampling technique, thereby eliminating the need for additional hyperparameters.

More evaluation against adaptive attacks. While the current evaluation includes a gray-box scenario, a more comprehensive evaluation against a broader range of adaptive attacks is crucial for a comprehensive understanding of the robustness of our method. In particular, the specific mechanisms of our approach, such as occlusion sensitivity maps, may present potential vulnerabilities that adaptive adversaries could exploit to generate more effective adversarial examples. It is possible to implement an adaptive attack that remains effective even after masking occlusion sensitivity maps (by inserting malicious perturbations into unimportant regions) and evaluate the robustness under this attack. Testing the full spectrum of adversarial strategies, particularly advanced adaptive attacks, is crucial to improving the resilience of our method against such threats.

REFERENCES

[1] T. A. Nguyen and A. T. Tran. 'WaNet - Imperceptible Warping-based Backdoor Attack'. In: *International Conference on Learning Representations*. 2021.

[2] 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.

[3] 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. 11966–11976.

[4] K. D. Doan, Y. Lao and P. Li. 'Marksman Backdoor: Backdoor Attacks with Arbitrary Target Class'. In: *arXiv preprint arXiv:2210.09194* (2022).

[5] T. Wang, Y. Yao, F. Xu, S. An, H. Tong and T. Wang. 'An invisible black-box backdoor attack through frequency domain'. In: *European Conference on Computer Vision*. Springer. 2022, pp. 396–413.

[6] X. Han, Y. Wu, Q. Zhang, Y. Zhou, Y. Xu, H. Qiu, G. Xu and T. Zhang. 'Backdooring multimodal learning'. In: *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2024, pp. 3385–3403.

[7] S. Liang, M. Zhu, A. Liu, B. Wu, X. Cao and E.-C. Chang. 'BadCLIP: Dual-Embedding Guided Backdoor Attack on Multimodal Contrastive Learning'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2024, pp. 24645–24654.

[8] C. Luo, Y. Li, Y. Jiang and S.-T. Xia. 'Untargeted backdoor attack against object detection'. In: *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2023, pp. 1–5.

[9] H. Zhang, Y. Wang, S. Yan, C. Zhu, Z. Zhou, L. Hou, S. Hu, M. Li, Y. Zhang and L. Y. Zhang. 'Test-time backdoor detection for object detection models'. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2025, pp. 24377–24386.

[10] S. Cheng, G. Shen, G. Tao, K. Zhang, Z. Zhang, S. An, X. Xu, Y. Li, S. Ma and X. Zhang. 'Odscan: Backdoor scanning for object detection models'. In: *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2024, pp. 1703–1721.

[11] S.-Y. Chou, P.-Y. Chen and T.-Y. Ho. 'How to Backdoor Diffusion Models?' In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2023, pp. 4015–4024.

[12] W. Chen, D. Song and B. Li. ‘TrojDiff: Trojan Attacks on Diffusion Models with Diverse Targets’. In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 4035–4044.

[13] D. Liu, Y. Qiao, R. Wang, K. Liang and G. Smaragdakis. ‘LADDER: Multi-objective Backdoor Attack via Evolutionary Algorithm’. In: *arXiv preprint arXiv:2411.19075* (2024).

[14] Y. Shi, M. Du, X. Wu, Z. Guan, J. Sun and N. Liu. ‘Black-box Backdoor Defense via Zero-shot Image Purification’. In: *Advances in Neural Information Processing Systems*. 2023, pp. 57336–57366.

[15] C. Xie, K. Huang, P.-Y. Chen and B. Li. ‘Dba: Distributed backdoor attacks against federated learning’. In: *International Conference on Learning Representations*. 2019.

[16] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee and D. Papailiopoulos. ‘Attack of the tails: Yes, you really can backdoor federated learning’. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 16070–16084.

[17] 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. 11966–11976.

[18] Z. Zhao, X. Chen, Y. Xuan, Y. Dong, D. Wang and K. Liang. ‘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, pp. 15213–15222.

[19] M. S. Ozdayi, M. Kantarcioglu and Y. R. Gel. ‘Defending against backdoors in federated learning with robust learning rate’. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 10. 2021, pp. 9268–9276.

[20] H. Zhang, J. Jia, J. Chen, L. Lin and D. Wu. ‘A3fl: Adversarially adaptive backdoor attacks to federated learning’. In: *Advances in Neural Information Processing Systems* 36 (2024).

[21] M. Choe, C. Park, C. Seo and H. Kim. ‘SDBA: A Stealthy and Long-Lasting Durable Backdoor Attack in Federated Learning’. In: *arXiv preprint arXiv:2409.14805* (2024).

[22] E. Luhman and T. Luhman. ‘Knowledge distillation in iterative generative models for improved sampling speed’. In: *arXiv preprint arXiv:2101.02388* (2021).

[23] M. T. Ribeiro, S. Singh and C. Guestrin. ‘" Why should i trust you?" Explaining the predictions of any classifier’. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144.

ACKNOWLEDGEMENTS

As I stand at the culmination of this incredible PhD journey, I am grateful to my supervisors, friends, colleagues and family, for their support, guidance, encouragement and inspiration throughout my PhD studies. This achievement would not have been possible without their contributions, which have been important in shaping my academic and personal growth.

First and foremost, I would like to express my deepest appreciation to my promoters, Prof. Inald Lagendijk and Dr. Kaitai Liang, for their guidance, patience, and mentorship. To Inald, thank you for all the insights and feedback in our six-weekly meetings. Your ability to challenge me with the fundamental question of "why" has not only deepened my understanding of the underlying mechanisms beyond empirical findings but also enhanced my critical thinking and analytical skills. Your professional mentorship has inspired me to explore the complexities of research with rigor and curiosity, shaping me into a more thoughtful and reflective scholar. Thank you for also guiding me through my propositions they challenge me to think deeply. To Kaitai, thank you for all the supports and guidance in my entire PhD life. Your meticulous review of my papers, along with your constructive comments, has greatly improved my scientific writing and logical reasoning abilities. I vividly recall the countless nights spent together refining manuscripts and navigating the challenges of peer review. Without your dedication and expertise, the publication of high-quality papers would not have been possible. Your criticism has been instrumental in helping me recognize and address my shortcomings, while your encouragement has been a source of strength during the most challenging moments of my PhD journey. I am so lucky to have the opportunity to work with you over the past four years, and I am deeply appreciative of everything you have done to support my academic and personal growth. I sincerely wish you both continued success in your professional endeavors and hope that your future contributions will inspire and guide many more PhD students, just as you have inspired me.

I would like to extend my sincere gratitude to my committee members, Prof.dr.ir. R.L. Lagendijk, Dr. K. Liang, Prof.dr. C.D. Jensen, Dr. A. Nocera, Prof.dr. M.E. van Dijk, Dr.ir. S. Verwer, Prof.dr. G. Smaragdakis, for their invaluable time and effort in reviewing my thesis and participating in my PhD defense ceremony. I am also grateful to Prof.dr.ir. M.J.T. Reinders for serving as the reserve committee member for my doctoral defense.

I would like to express my heartfelt gratitude to my friends for their companionship and support throughout my PhD journey. First, I extend my sincere thanks to Rui and Huanhuan for their support when I first arrived in Delft. Rui, thank you for your help with coding and your prompt responses to my numerous questions. Huanhuan, thank you for your insightful discussions from a professional cryptography perspective, as well as many travel suggestions. Our hiking adventures exploring the Netherlands together were unforgettable and filled with joy. Tianyu and Zeshun, thank you for organizing and

hosting amazing party nights. I thoroughly enjoyed every evening spent playing board games and cards with you. Tianyu, I am also grateful for your efforts in organizing trips to explore the breathtaking landscapes of Europe. To Jinke, Jing, Huimin, Shihui, and Hao, thank you for small talks, nice dinners, and wonderful board games. Your friendship has been a source of great comfort and joy. A special thanks to Dazhuang for your help with both small and big things during my PhD journey. I appreciate and enjoy our valuable research discussions. I wish you a successful completion of your PhD journey with many high-quality publications. Finally, I feel fortunate to have met all of you during my PhD journey. Your companionship has made this experience truly memorable and enriching.

To my colleagues in the CYS Group, thank you for creating such a safe, supportive and collaborative environment. I would like to thank George and Sandra for their efforts in organizing numerous engaging activities, CYS outings, and coffee breaks. Also, I would like to thank Harm, Lilika, Giovane, Alexios, Zeki, Roland, Jelle, Daniël, Clinton, Florine, Jorrit, Stefanos, Marwan, Misha, and Tjard for the coffee breaks and happy moments we shared. Moreover, I would like to thank Andrea for the insightful discussions. Finally, I would like to thank several excellent MSc students under my supervision, Armin, Congwen and Andrei. Thank you for your dedication and hard work. It was a pleasure to collaborate with you on your thesis projects, and I am proud of what we accomplished together.

To my family, words cannot express how thankful I am for your endless love and sacrifices. Your weekly video calls keep me from feeling lonely in a foreign land. Your belief in me has been a constant source of encouragement, and I am forever grateful for your love and support, even from afar. You will always be the ones who comfort me when I face setbacks. A special thanks to my girlfriend, Hui, for your understanding and companionship. The decision to pursue a PhD in the Netherlands has cost me much time that I could have spent with you. Thank you for your support again.

This dissertation is not just a reflection of my efforts, but a testament to the collective support and help of so many. The dissertation is dedicated to the memory of my grandfather, a guiding light and enduring inspiration in my life. Finally, to all those who have contributed to this wonderful journey in big and small ways, thank you.

Yanqi Qiao
Delft, February 2025

CURRICULUM VITÆ

Yanqi Qiao was born in Manzhouli, Inner Mongolia, China. He obtained his bachelor's degree in Information Security from Wuhan University, Wuhan, China in 2018. After that, he received his master's degree in Cyber Security at Wuhan University, Wuhan, China in 2021.

In 2021, he started as a Ph.D student at Delft University of Technology, Delft, the Netherlands under the supervision of Prof.dr.ir.R.L.Lagendijk and Dr. Kaitai Liang. He supervised multiple master students during their final thesis projects.

During his Ph.D., he worked on exploring the vulnerabilities of different machine learning models and enhancing their security, especially in computer vision tasks. He has published a series of research works, applying machine learning tools to address cybersecurity challenges. These publications have appeared in high-tier international security conferences, e.g., NDSS.

List of Publications

Journal

2. **Qiao, Y.**, Liu, D., Wang, R., Liang, K., 2024. Stealthy Backdoor Attack against Federated Learning through Frequency Domain by Backdoor Neuron Constraint and Model Camouflage. In IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS).

1. Stenhuis, R., Liu, D., **Qiao, Y.**, Conti, M., Panaousis, M. and Liang, K., 2025. MeetSafe: Enhancing Robustness against White-box Adversarial Examples. In Frontiers in Computer Science.

Conference

3. Liu, D.*, **Qiao, Y.***, Wang, R., Liang, K., Smaragdakis, G., 2025. LADDER: Multi-objective Backdoor Attack via Evolutionary Algorithm. In the Network and Distributed System Security (NDSS) Symposium.

2. **Qiao, Y.**, Liu, D., Wang, R., Liang, K., 2025. Low-Frequency Black-Box Backdoor Attack via Evolutionary Algorithm. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV).

1. Tian, Y., Wang, R., **Qiao, Y.**, Panaousis, E. and Liang, K., 2023. FLVoogd: Robust and Privacy Preserving Federated Learning. In Asian Conference on Machine Learning (ACML).

Preprint

4. Liu, D., **Qiao, Y.,** Wang, R., Liang, K., Smaragdakis, G., 2025. PASTA: A Patch-Agnostic Twofold-Stealthy Backdoor Attack on Vision Transformers. Under review.

3. **Qiao, Y.**, Liu, D., Panaousis, M., Conti, M. and Liang, K., 2023. FTA: Stealthy Backdoor Attack with Flexible Triggers against Federated Learning. arXiv preprint arXiv:2309.00127.

2. Popovici, A., **Qiao, Y.**, Liu, D., Smaragdakis, G. and Liang, K., 2025. MSID: Multi-Scale Diffusion-Based Inpainting Defense Against Adversarial Attacks. Under review.

1. Amalan, A., Wang, R., **Qiao, Y.**, Panaousis, E. and Liang, K., 2022. MULTI-FLGANs: Multi-Distributed Adversarial Networks for Non-IID distribution. arXiv preprint arXiv:2206.12178.


TUDelft
9 789464 739565