ECAI 2025
I. Lynce et al. (Eds.)

doi:10.3233/FAIA250939

IPBA: Imperceptible Perturbation Backdoor Attack in Federated Self-Supervised Learning

Jiayao Wang[a], Yang Song[a], Zhendong Zhao[b], Jiale Zhang[a], Junwu Zhu[a,*], Qilin Wu[c] and Dongfang Zhao[d]

[a]School of Information Engineering, Yangzhou University, China
[b]Institute of Information Engineering, Chinese Academy of Sciences, China
[c]School of Computing and Artificial Intelligence, Chaohu University, China
[d]Tacoma School of Engineering and Technology, University of Washington, USA

Abstract. Federated Self-Supervised Learning (FSSL) combines the advantages of decentralized modeling and unlabeled representation learning, serving as a cutting-edge paradigm with strong potential for scalability and privacy preservation. Although FSSL has garnered increasing attention, research indicates that it remains vulnerable to backdoor attacks. Existing methods generally rely on visually obvious triggers, which makes it difficult to meet the requirements for stealth and practicality in real-world deployment. In this paper, we propose an imperceptible and effective backdoor attack method against FSSL, called IPBA. Our empirical study reveals that existing imperceptible triggers face a series of challenges in FSSL, particularly limited transferability, feature entanglement with augmented samples, and out-of-distribution properties. These issues collectively undermine the effectiveness and stealthiness of traditional backdoor attacks in FSSL. To overcome these challenges, IPBA decouples the feature distributions of backdoor and augmented samples, and introduces Sliced-Wasserstein distance to mitigate the out-of-distribution properties of backdoor samples, thereby optimizing the trigger generation process. Our experimental results on several FSSL scenarios and datasets show that IPBA significantly outperforms existing backdoor attack methods in performance and exhibits strong robustness under various defense mechanisms.

1 Introduction

In recent years, Self-Supervised Learning (SSL) [11, 3, 2, 8] has emerged as a powerful paradigm in machine learning, particularly in computer vision. The main advantage of SSL lies in its ability to learn rich representations from large amounts of unlabeled data, bypassing the labor-intensive and costly manual labeling process. SSL in computer vision aims to develop image encoders that produce similar embeddings for similar images. To achieve this, similar image pairs are typically constructed by applying various augmentations to the same image. The pre-trained encoder can then be used to train downstream classifiers for various tasks. These downstream classifiers generally use compact networks with fewer parameters, improving training efficiency and reducing computational cost.

Although SSL has made significant progress with unlabeled data, it requires vast amounts of data to achieve performance comparable to supervised learning. This substantial data requirement can be overwhelming for individuals or organizations and may become a significant barrier to model training in data-scarce domains. In this context, Federated Self-Supervised Learning (FSSL) [39, 22] employs a distributed learning approach, providing an effective alternative to the data-sharing problem. Modern edge devices have access to vast amounts of data suitable for training models, giving FSSL a significant advantage. However, the decentralized nature of traditional Federated Supervised Learning (FSL) makes it vulnerable to backdoor attacks [1, 20], and this vulnerability may be inherited in FSSL, posing a potential security risk.

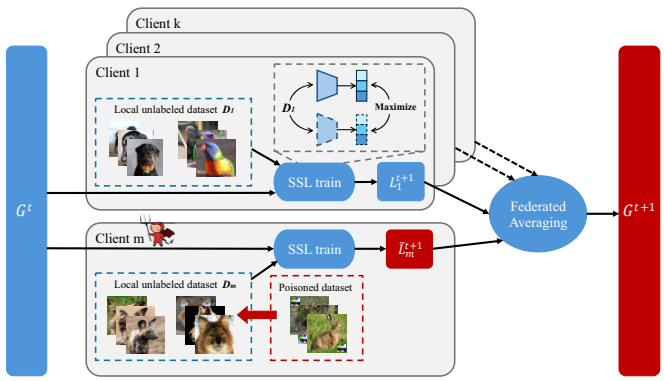

Figure 1. Backdoor injection process in federated self-supervised learning.

Recent studies [37, 21, 33] have shown that FSSL is vulnerable to backdoor attacks. Figure 1 illustrates the backdoor injection process in FSSL. During this process, malicious clients inject backdoors into the global model by training their local models with poisoned samples that include triggers. Specifically, benign clients $i \in \{1, 2, \ldots, N\}$ use local unlabeled datasets D_i for self-supervised learning, and the Federated Averaging (FedAvg) mechanism aggregates their updated local model L_i^{t+1} to the central server to update the global model G^{t+1}. Meanwhile, malicious clients (e.g., client m) train on poisoned data, generating backdoored models $\tilde{L}_m^{t+1}$, which are also uploaded to the server. Through the FedAvg, all client models (including those from malicious clients with backdoors) are aggregated into the global model G^{t+1}, embedding the backdoor into the updated global model. When the global encoder is used to construct downstream tasks, the backdoored classifier predicts each input embedded with the attacker's selected trigger as the corresponding target class specified by the attacker.

* Corresponding Author. Email: jwzhu@yzu.edu.cn

However, a common drawback of these existing attack methods is that their trigger patterns are highly visible, making them vulnerable to detection through manual inspection or defense mechanisms. In this paper, we propose a backdoor attack in FSSL that is both effective and stealthy.

A feasible approach is to apply existing imperceptible triggers [19, 20, 16] for use in the FSSL setting. However, experimental results indicate that these imperceptible triggers, which were originally designed for Supervised Learning (SL), exhibit limited effectiveness when applied to SSL models. Our observations suggest that the primary reason for this ineffectiveness lies in the feature distribution entanglement between backdoor samples and the augmented samples used in FSSL. Specifically, the transformations induced by the backdoor triggers share similarities with the image augmentations inherent to contrastive learning in FSSL (such as RandomGrayscale and ColorJitter). As a result, local SSL models struggle to distinguish between the feature distributions of backdoor samples and augmented samples.

Building on the above observations, we propose an imperceptible perturbation backdoor attack that decouples the feature distributions of backdoor samples and augmented samples. Specifically, this approach involves increasing the distributional gap between the backdoor samples and augmented samples during the local SSL process, thereby enhancing their separability. Additionally, to further ensure the stealthiness of the trigger, IPBA imposes a distance constraint on the backdoor samples in the feature space using Sliced Wasserstein Distance [14], effectively reducing the out-of-distribution properties of the backdoor samples.

Our main contributions are summarized as follows:

- **We propose an imperceptible perturbation backdoor attack method:** In FSSL, we propose an innovative backdoor attack method, IPBA. By decoupling the feature distributions of backdoor samples and augmented samples, this method significantly enhances both the stealthiness and effectiveness of backdoor attacks.
- **We propose to apply Sliced Wasserstein Distance:** To mitigate the out-of-distribution properties of backdoor samples in the feature space, we innovatively introduce Sliced Wasserstein Distance. This approach effectively reduces the outlier phenomenon of backdoor samples by minimizing the distance between backdoor and clean samples in the feature space.
- **Extensive experimental evaluation:** We conduct comprehensive evaluations of the proposed method on five public benchmark datasets (CIFAR10, STL10, GTSRB, SVHN, and Tiny-ImageNet). The experimental results show that our method significantly outperforms existing backdoor attack methods in terms of performance and demonstrates strong generalization across various settings. Additionally, we further explore potential defense strategies against IPBA and find that current state-of-the-art defense methods have limitations, emphasizing the urgent need for tailored defense mechanisms.

2 Related Work

2.1 Federated Self-Supervised Learning

Federated self-supervised learning has gained increasing attention due to its ability to jointly learn representations across decentralized clients without relying on labeled data, while preserving data privacy. Early studies [29] explored the direct integration of classical SSL methods such as SimCLR and BYOL into the federated setting. To address data heterogeneity, approaches like SSFL [9] and FedEMA [39] introduced personalization and momentum-based model updates, respectively. Other efforts, such as FedCA [34] and L-DAWA [22], focused on enhancing model aggregation via global dictionary learning or divergence-aware weighting. However, existing research primarily focuses on performance optimization and modeling data heterogeneity, while overlooking potential backdoor risks.

2.2 Backdoor Attacks

Attackers conducting backdoor attacks typically select a stealthy trigger and embed it into a subset of training samples to poison the training data. Traditional backdoor techniques rely on explicit labels [20], manipulating sample labels to guide the model toward learning attacker-specified behaviors. However, the absence of labeled data in SSL renders traditional backdoor paradigms inapplicable, thereby prompting the emergence of novel methodologies for backdooring SSL. BASSL [25] embeds triggers into target-class images and leverages cropping-based augmentations to generate diverse poisoned views. BadEncoder [13] fine-tunes a pre-trained encoder using a trigger-injected shadow dataset to precisely steer model behavior. Additionally, CorruptEncoder [36] and PoisonedEncoder [17] respectively propose poisoning strategies that target the victim's training dataset. Recent studies have demonstrated that FSSL is also susceptible to backdoor threats [37, 33, 21]. For example, BADFSS [37] injects backdoor triggers into the global encoder by leveraging supervised contrastive learning and attention alignment. However, existing methods typically rely on visually perceptible triggers, making them easily detectable by humans or automated detection systems. In contrast, our approach surpasses existing methods in both stealth and effectiveness.

2.3 Backdoor Defenses

Backdoor attacks have attracted considerable attention due to their high stealthiness and potential for severe damage, which makes it a challenge to effectively defend against such attacks during model training. On the one hand, reverse engineering-based methods (e.g., Neural Cleanse [30] and DECREE [6]) aim to reconstruct the trigger from a backdoored model and identify its corresponding target class. Successful trigger reconstruction typically indicates that the encoder is compromised. On the other hand, sample-level detection methods identify anomalies by analyzing the influence of input samples on model predictions. For example, STRIP [7] measures prediction consistency under input perturbations to detect potential poisoned samples, while GradCAM [26] utilizes activation map visualization to localize trigger regions and assist in identifying backdoored inputs. We use the above state-of-the-art defense techniques to evaluate our newly proposed attack.

3 Observations and Intuitions

Observation I: Limited transferability. Existing imperceptible triggers are not as effective as expected in the FSSL environment. As shown in Figure 2, we present the Attack Success Rate (ASR) of existing imperceptible triggers on the FSSL model. We transferred the comparative methods to the FSSL scenario, using the attack methods introduced in BADFSS [37] and replacing the original patch triggers with imperceptible triggers. The results show that although these imperceptible triggers achieve high ASR in SL and SSL, they do not perform well in the FSSL environment.

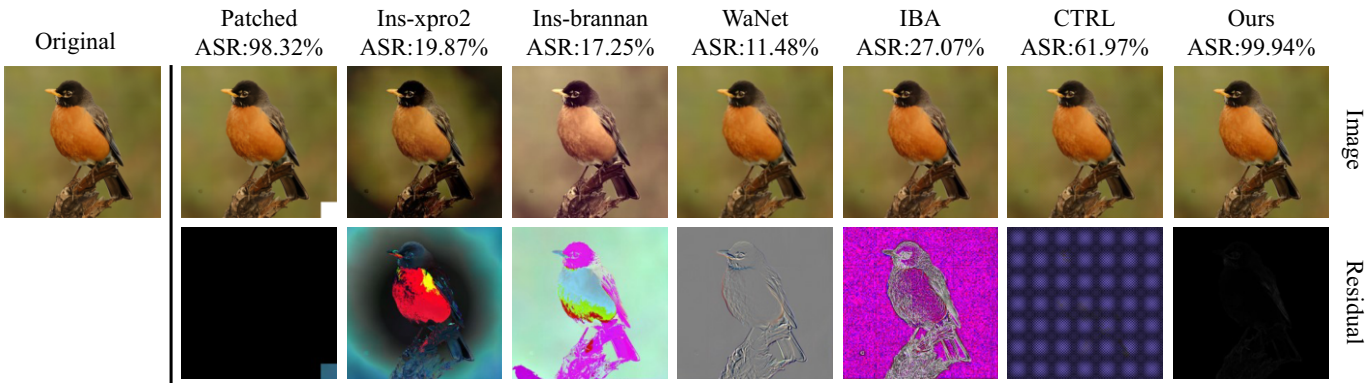

Figure 2. Comparison of clean, backdoored samples created by Patch trigger, Instagram filter trigger, WaNet trigger, IBA trigger, CTRL trigger and ours. The ASR under the BADFSS threat model is shown next to each method name. Residuals are the difference between clean and backdoored images.

Observation II: Feature entanglement. The augmented samples are entangled with the backdoor samples in the feature distribution. Specifically, we use a pre-trained ResNet18 [10] for binary classification to determine whether IBA-injected backdoor samples and contrastively augmented samples are distinguishable. Experimental results indicate that the model struggles to differentiate between the two types of samples. The t-SNE visualization in the left panel of Figure 3 shows significant overlap between the two types of samples in the feature space. Based on this observation, we infer that the reduced effectiveness of supervised backdoor attacks in FSSL may be attributed to the high similarity in feature distribution between contrastively augmented samples and backdoor samples, which weakens the model's ability to distinguish between the two. In contrast, our method (right panel of Figure 3) effectively decouples the feature distributions of poisoned and augmented samples in the latent space.

Observation III: Out-of-distribution property. Existing backdoor samples exhibit out-of-distribution properties. Tao et al. [28] present a critical insight that current SSL attacks introduce strong backdoor signals into the embedding space, pushing malicious samples outside the clean data distribution. Inspired by this insight, we conducted experiments to verify whether a similar phenomenon occurs in FSSL. Specifically, we fed both clean pre-training data and poisoned samples into the same encoder to extract their feature embeddings. Then, Principal Component Analysis (PCA) was applied to reduce the dimensionality of these embeddings for visualization in a two-dimensional space. As shown in the left panel of Figure 4, existing FSSL attacks also exhibit such out-of-distribution properties.

Intuition and Design Motivation. Based on the above observations (more results are provided in the Appendix [31]) which highlight challenges such as limited transferability, feature entanglement, and out-of-distribution properties, we identify several key factors that are essential for achieving stealthy backdoor attacks in FSSL:

- Decoupling the feature distributions of backdoor samples and augmented samples during local client pre-training. Since the augmentation strategies in the pre-training stage are known in advance, we design $\mathcal{L}_{dis}$ to quantify the distributional gap between the two batches of images.
- Introducing a dual alignment loss tailored for malicious clients. To further enhance the effectiveness of the backdoor attack, we design $\mathcal{L}_{align}$ to pull close the features of backdoor images and target images.
- An excessive distributional gap may reduce the visual naturalness of backdoor images and compromise the stealthiness of the trigger. Therefore, we design $\mathcal{L}_{ste}$ to seamlessly fuse the backdoor with the original image and eliminate out-of-distribution properties in the feature space.

4 Methodology

In this section, we first introduce preliminaries of FSSL system model, Wasserstein Distance, and threat model. We then elaborate on the design of the Poisoned Data Constructor phase. Finally, we describe the Backdoor Injection strategy and formulate its corresponding optimization objective.

4.1 Threat Model

Attack Objective. In FSSL, the attacker's goal is to inject a backdoor into the global model while maintaining both effectiveness and stealthiness. Effectiveness means the model consistently misclassifies trigger inputs into a target class, yielding high ASR. Stealthiness means the backdoor does not noticeably degrade the model's main-task performance, thus evading detection. Overall, the attacker aims to induce the global SSL model to exhibit malicious behavior on specific inputs, without significantly affecting its overall utility or being exposed during training.

Attack Knowledge and Capabilities. We assume the attacker masquerades as a benign FL participant, with knowledge of the global model and full control of local training. This includes the ability to manipulate the local dataset (e.g., by embedding triggers), redesign the loss function, and arbitrarily alter the model training procedure. Therefore, the attacker also has access to the data augmentation strategies used for pretraining the encoder, which can be leveraged in IPBA to generate stealthy and effective backdoor triggers. However, the attacker is generally unaware of other critical details of the FSSL system, such as the models from benign clients and the aggregation rules employed by the server.

4.2 Preliminaries

FSSL System Model. We consider a standard FSSL framework, in which a total of N decentralized clients collaboratively train a global encoder $\mathbf{W}_{global}$. In each communication round, every selected client updates its local model using its private unlabeled dataset $\mathcal{D}_i = \{\mathbf{x}_1, \ldots, \mathbf{x}_{|\mathcal{D}_i|}\}$, where $i \in \mathcal{N} = \{1, 2, \ldots, N\}$.

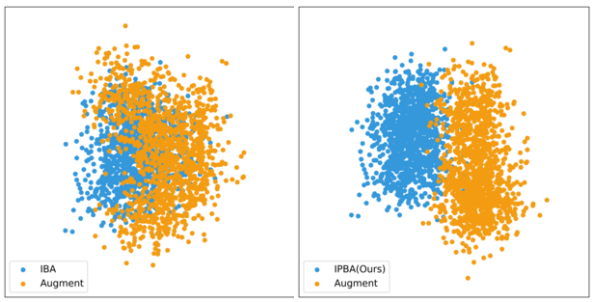

Figure 3. The t-SNE visualization of feature vectors in the latent space under different attacks.

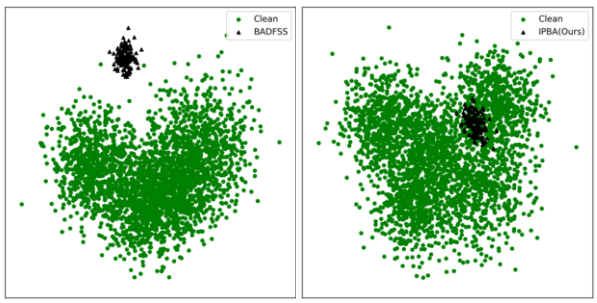

Figure 4. PCA visualization of clean and poisoned sample embeddings in backdoored models under different attacks.

The training proceeds iteratively through the following three steps:

Step 1: Global Model Distribution. At the beginning of round t, the central server broadcasts the current global encoder $\mathbf{W}^t_{global}$ to a subset of selected clients $\mathcal{S}^t$.

Step 2: Local Self-Supervised Training. Each client $i \in \mathcal{S}^t$ initializes its local model with the received global encoder $\mathbf{W}^t_{global}$, and performs e local training epochs on its unlabeled dataset $\mathcal{D}_i$. The resulting updated local encoder is denoted by $\mathbf{W}^t_i$, which is then uploaded to the central server.

Step 3: Federated Aggregation. Upon receiving all local updates, the server aggregates them to obtain a new global encoder $\mathbf{W}^{t+1}_{global}$. Following the FedAVG protocol, the aggregation is performed as: $\mathbf{W}^{t+1}_{global} = \sum_{i\in\mathcal{S}^t} p^t_i \mathbf{W}^t_i$, where the aggregation weight is given by $p^t_i = \frac{|\mathcal{D}_i|}{\sum_{j\in\mathcal{S}^t}|\mathcal{D}_j|}$.

After finishing the training process, we freeze the parameters of the global model, and use it to construct a downstream predictor with only a small amount of labeled data.

Wasserstein Distance. Wasserstein Distance (WD) is a widely used metric for measuring the discrepancy between underlying distributions, particularly in scenarios where the common support or density functions are unknown. As our goal is to minimize the distributional difference between poisoned and clean samples, Wasserstein distance offers a smooth and effective solution to this problem. We adopt the 2-Wasserstein distance, also known as the Earth Mover's Distance, which is defined as:

$$\mathcal{W}(\zeta,\tau) = \left(\inf_{\psi\in\Pi(\zeta,\tau)} \int_{(u,v)\sim\psi} p(u,v)\,\|u-v\|_2\, dudv\right)^{1/2}, \quad (1)$$

where ζ and τ represent the marginal probability distributions of clean and poisoned samples in the feature space, estimated from their corresponding empirical embeddings. ψ denotes the joint distribution between the two. The infimum $\inf$ denotes the greatest lower bound of the computed distance on the joint distribution. The integral term calculates the sum of the distance between every two data points (u, v) drawn from the joint distribution ψ. $p(u, v)$ is the probability of jointly drawing the two samples. $||u - v||_2$ is the L^2 distance between the two samples.

4.3 Poisoned Data Constructor

In the FSSL framework, when a malicious client is selected to participate in a training round, it gains the opportunity to influence the global model update. Specifically, malicious client k receives the current global parameters $\mathbf{W}^t_{global}$, updates them using its local dataset $\mathcal{D}_k$, and obtains a locally refined model $\mathbf{W}^t_k$. This local model is subsequently incorporated into the next-round global model through the FedAvg aggregation scheme. Based on this process, we introduce an initial phase termed the Poisoned Data Constructor, in which a trainable backdoor injector $\mathcal{I}_\psi$ is designed to transform a clean input x into a poisoned sample $x' = \mathcal{I}_\psi(x)$. This sample remains imperceptibly different from the original, but it effectively misleads the model into making incorrect classifications for the target class. To this end, we design three essential loss functions that jointly optimize the poisoned samples in terms of effectiveness, disentanglement, and stealthiness.

Dual Alignment Loss. Given that FSSL models generate feature embeddings rather than relying on explicit labels, a malicious client must optimize trigger-injected inputs to align with the target semantics in order to achieve attack effectiveness. Moreover, as a critical step toward enhancing attack efficiency, the attacker must ensure that its local model can accurately recognize the target semantics. We refer to this process as dual feature alignment between the backdoor sample and the target sample. Following [13], the dual feature alignment of the malicious client can be formally expressed as:

$$\mathcal{L}_{align} = -\frac{1}{|\mathcal{D}_k|}\sum_{x\in\mathcal{D}_k}\Big[s\left(f(x',\tilde{\theta}), f(x_t,\tilde{\theta})\right) - s\left(f(x_t,\tilde{\theta}), f(x_t,\theta)\right)\Big], \quad (2)$$

where $s(\cdot,\cdot)$ is the cosine similarity function, $f(\cdot,\theta)$ denotes the feature encoder, x' is the poisoned sample generated by the injector $\mathcal{I}_\psi$, x_t is the target class sample, and $\tilde{\theta}, \theta$ are the backdoored and clean model parameters, respectively.

Feature Disentanglement Loss. To mitigate the severe overlap between poisoned samples and their augmented counterparts in the feature space, we design a feature disentanglement loss, inspired by [35]. This loss maximizes the representational distance between the two, encouraging the encoder to learn more separable feature embeddings in the latent space, thereby enhancing the recognizability and robustness of the backdoor semantics. Specifically, we first convert the images from the RGB color space to HSV and HSL, as these color representations better capture the variations introduced by augmentation operations such as ColorJitter, which randomly perturbs brightness, contrast, saturation, and hue. Then, we compute the differences between the poisoned images and their augmented counterparts in both color spaces, based on which the following loss function is defined:

$$\mathcal{L}_{dis} = \mathbb{E}_{x\sim\mathcal{D}_k}\left[\sum_{c\in\{H,S,V,L\}}\left\|c(x') - c(\tilde{x})\right\|_2^2\right], \quad (3)$$

where H, S, V, L represent the Hue, Saturation, Value, and Lightness channel transformations from the HSV and HSL color spaces,

respectively. The input sample x augmented by the transformations used during the encoder's pre-training stage is denoted as $\tilde{x}$. The function $c(\cdot)$ denotes the color channel transformation for a specific channel (H, S, V, L), and $\|u - v\|_2^2$ denotes the squared ℓ_2 distance between sample u and sample v.

Stealthiness Loss. To address the out-of-distribution characteristics of poisoned samples, we introduce a stealthiness loss aimed at minimizing the distributional discrepancy between poisoned and clean samples. The 2-Wasserstein distance serves as an effective metric for this purpose. However, directly computing WD in high-dimensional spaces is challenging due to the involved optimization process. To overcome this, we adopt the Sliced Wasserstein Distance (SWD) [14], a variant that projects high-dimensional data onto multiple random one-dimensional subspaces, computes WD in each, and averages the results. This approach significantly improves computational efficiency and numerical stability, as the 1D 2-Wasserstein distance admits a closed-form solution.

$$\mathcal{W}_{sliced}(\zeta, \tau) = \left(\frac{1}{S} \sum_{s=1}^{S} \int_0^1 \|F_c^s(z) - F_b^s(z)\|_2 \, dz \right)^{\frac{1}{2}}, \quad (4)$$

where S is the number of one-dimensional directions. F_c^s and F_b^s represent the projections of the clean and poisoned embeddings into one-dimensional data points along the direction of slice s, respectively.

We first extract high-dimensional feature representations of poisoned and clean samples using a pre-trained model F, and then employ the SWD to effectively measure the distributional discrepancy between them. The final stealthiness loss is defined as follows:

$$\mathcal{L}_{ste} = \sum_{x \in \mathcal{D}_k} \text{SWD}\left(F(x'), F(x)\right). \quad (5)$$

Based on the above loss formulations and design objectives, we define the optimization of the backdoor injector $\mathcal{I}_\psi$ as a joint learning task:

$$\arg\min_{\mathcal{I}_\psi} \mathcal{L}_{injector} = \mathcal{L}_{ste} + \alpha \cdot \mathcal{L}_{dis} + \beta \cdot \mathcal{L}_{align}, \quad (6)$$

where α and β are hyperparameters used to balance the contributions of the three loss terms.

4.4 Backdoor Injection

The core objective of the second-stage backdoor injection is to guide the model to learn backdoor behaviors through poisoned data during local training and progressively migrate these behaviors into the global model. Specifically, each selected malicious client is required to embed backdoor features into its local model while maintaining stealthiness. To achieve this, the attacker utilizes trigger-embedded samples to construct feature representations that are semantically similar to the target class, thereby inducing the model to produce the desired responses to backdoor inputs. In addition, to enhance the effectiveness and efficiency of the attack, the attacker guides the backdoored model to produce feature representations for reference data that are similar to those generated by the clean model, thereby ensuring accurate recognition of target-class samples by the backdoor model (as shown in Equation (2)).

Utility Loss. To ensure that the injection process does not significantly degrade the performance on the main task, the attacker must maintain the local model's ability to represent clean samples correctly, such that its output features remain consistent with those of the clean model. Based on this, we define the utility loss $\mathcal{L}_{uti}$ as follows:

$$\mathcal{L}_{uti} = -\frac{1}{|\mathcal{D}_k|} \sum_{x \in \mathcal{D}_k} s\left(f(x, \tilde{\theta}), f(x, \theta)\right). \quad (7)$$

Combining $\mathcal{L}_{align}$ and $\mathcal{L}_{uti}$, the objective of local encoder backdoor injection can be formulated as the following optimization problem:

$$\arg\min_{\tilde{\theta}} \mathcal{L}_{encoder} = \lambda_1 \cdot \mathcal{L}_{align}(\tilde{\theta}) + \lambda_2 \cdot \mathcal{L}_{uti}(\tilde{\theta}), \quad (8)$$

where λ_1 and λ_2 are hyperparameters that balance the two loss components.

5 Evaluation

To demonstrate the effectiveness and stealthiness of our approach, we implemented IPBA using Pytorch and compared its performance with existing state-of-the-art backdoor attack methods. All experiments were conducted on an NVIDIA 4090 GPU. We designed comprehensive experiments to address the following three research questions:

RQ1 (Effectiveness of IPBA): Can IPBA successfully inject backdoors into FSSL?

RQ2 (Stealthiness of IPBA): Can IPBA achieve good stealthiness and naturalness across different evaluation metrics?

RQ3 (Robustness of IPBA): Can IPBA effectively resist existing defense methods?

5.1 Experimental Setup

Datasets. Five datasets are employed in the experiments including CIFAR-10 [15], STL-10 [4], GTSRB [27], SVHN [18], and Tiny-ImageNet [24]. More details about the used datasets can be found in Appendix [31].

Evaluation Metrics. Similar to existing work in [13, 28], we evaluated the effectiveness of all attack methods using three metrics: Clean Accuracy (CA), Attack Success Rate (ASR), and Backdoored Accuracy (BA). A well-executed backdoor attack should maximize the ASR while maintaining a high BA. More details are explained in Appendix [31]. To assess the stealthiness and naturalness of our IPBA, we used three metrics: SSIM [32], PSNR [12], and LPIPS [38]. In the experiments, higher SSIM and PSNR values, along with lower LPIPS, indicate better stealthiness and naturalness of the generated backdoored images.

Baseline. We compare IPBA with the state-of-the-art backdoor attack method, BADFSS [37], and use WaNet [19], IBA [20], and CTRL [16] as baseline triggers. These methods significantly outperform earlier backdoor attack methods in terms of stealthiness. Following the experimental setup in [37], we adapt all baseline methods to the FSSL scenario for evaluation and strictly follow the original implementations.

Implementation Details. We use SimCLR as the default self-supervised learning algorithm and employ ResNet-18 [10] as the default architecture network for the encoders. Moreover, we use a two-layer multi-layer perceptron (MLP) as a predictor. Following previous work [37, 13, 35], we set the decay rate m = 0.99, batch size B = 256, SGD as optimizer with learning rate lr = 0.001 and run experiments with K = 5 clients (one is malicious and the poison ratio is 1%) for E = 200 training rounds, where each client performs e = 3 local epochs in each round. We use the U-Net architecture [23] for the backdoor injector.

Table 1. Camparison of attack performance on different datasets. The best result are **highlighted**.

Pre-training Dataset	Downstream Dataset	Benign	WaNet [19]		IBA [20]		CTRL [16]		BADFSS [37]		Ours	
		CA	BA↑	ASR↑	BA↑	ASR↑	BA↑	ASR↑	BA↑	ASR↑	BA↑	ASR↑
CIFAR-10	STL-10	75.14	73.27	11.91	73.21	10.14	75.73	66.85	72.36	66.35	72.82	**96.11**
	GTSRB	82.84	77.16	7.71	77.36	35.41	77.53	62.07	75.32	70.21	78.64	**93.68**
	SVHN	63.52	56.86	12.45	58.67	30.91	60.35	45.91	65.64	56.38	71.03	**92.83**
STL-10	CIFAR-10	85.21	82.06	11.48	84.77	27.07	78.19	61.97	77.52	68.32	87.19	**99.94**
	GTSRB	76.32	79.84	4.41	81.65	17.29	70.37	61.28	72.32	69.93	74.83	**97.41**
	SVHN	56.47	55.74	15.24	55.07	14.13	53.77	49.36	54.97	54.66	60.42	**99.53**
Tiny-ImageNet	STL-10	89.58	87.14	12.60	87.51	10.44	82.26	47.14	75.71	51.29	87.15	**99.91**
	GTSRB	78.32	77.26	10.51	80.72	9.38	70.91	49.85	68.13	46.54	75.75	**96.91**
	SVHN	73.67	72.63	13.93	71.94	18.75	65.94	40.91	60.38	41.97	71.86	**95.25**

5.2 *Effectiveness Evaluation (RQ1)*

Effectiveness comparison with SOTA attack methods. To evaluate the effectiveness of IPBA, we compared its ASR and BA with four SOTA attack methods. The experiments followed a standard SSL setup, where the pre-training and downstream datasets were different. Table 1 shows the performance of different attack methods. The results indicate that IPBA achieves a high ASR while maintaining a high BA. Specifically, with STL-10 as the pre-training dataset and CIFAR-10 as the downstream dataset, IPBA achieved the best ASR (99.94%) and BA (87.19%). Compared to supervised-based attack methods (e.g., WaNet and IBA), IPBA outperforms them in terms of ASR across different downstream datasets, which validates our previous conclusion that supervised-based methods are unsuitable for the FSSL scenario. Furthermore, compared to self-supervised-based attack methods (e.g., CTRL and BADFSS), IPBA also shows superior performance in terms of ASR and BA across all datasets.

Effectiveness on different SSL algorithms. An effective attack method should exhibit strong generalizability and be adaptable to various SSL algorithms. To this end, we evaluate the performance of IPBA in FSSL under four representative SSL methods: SimCLR [3], MoCo [11], BYOL [8], and SwAV [8]. Figure 5 presents the attack performance of IPBA across these SSL algorithms. The results demonstrate that IPBA exhibits stable attack performance across different SSL algorithms, highlighting its adaptability and generalizability.

Effectiveness on different encoder architectures. To evaluate the effectiveness of IPBA across different encoder architectures, we conducted experiments on STL-10 using three representative architectures: ResNet-18 [10], ResNet-50 [10], and ViT [5]. Figure 5 illustrates the attack performance of IPBA on these encoder architectures. The results show that IPBA can successfully inject backdoors into various encoder architectures while maintaining high BA classification performance, highlighting its strong generalizability.

5.3 *Stealthiness Evaluation (RQ2)*

To evaluate the stealthiness of IPBA, we compared the trigger images generated by IPBA with those of SOTA attack methods. Additionally, we used PCA to visualize the embedding spaces of clean and poisoned samples on the backdoored models.

Stealthiness results from the perspective of images. Figure 2 compares the poisoned samples generated by IPBA with those from SOTA attack methods, along with their magnified residuals (×2). The results show that the residual generated by IPBA is the smallest, leaving only a few subtle artifacts.

Table 2. Stealthness evaluation on different datasets.

Pre-training Dataset	Downstream Dataset	SSIM↑	PSNR↑	LPIPS↓
CIFAR-10	STL-10	0.9142	22.19	0.0311
	GTSRB	0.9914	35.03	0.0017
	SVHN	0.9946	35.22	0.0012
STL-10	CIFAR-10	0.9898	35.38	0.0026
	GTSRB	0.9911	36.88	0.0041
	SVHN	0.9857	31.89	0.0032
Tiny-ImageNet	STL-10	0.9889	33.46	0.0044
	GTSRB	0.9908	32.68	0.0031
	SVHN	0.9846	31.57	0.0041

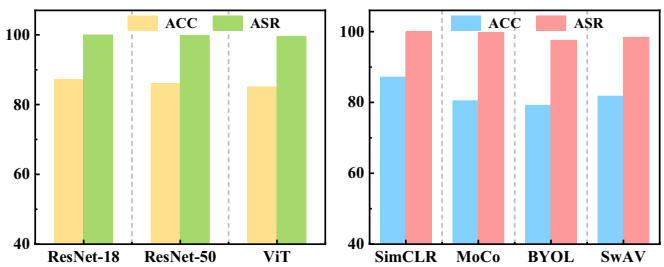

Figure 5. Experimental results for different encoder architectures and SSL algorithms.

Moreover, we evaluated the stealthiness of the triggers generated by IPBA using three metrics: PSNR, SSIM, and LPIPS. Table 2 presents the evaluation results across different datasets. As shown in the table, the SSIM values are consistently close to 1, with most exceeding 0.98, indicating that the structural changes introduced by IPBA are minimal. The PSNR values are notably high, suggesting that the noise introduced is nearly imperceptible. Additionally, the LPIPS values are extremely low, ranging from 0.0012 to 0.0311, further confirming that the perceptual difference between the original and perturbed images is negligible. Overall, these results demonstrate that IPBA effectively maintains the stealthiness of the images, introducing virtually no noticeable visual changes.

Stealthiness results from the perspective of latent space. Many backdoor defense methods assume that poisoned and benign samples are clearly separated in the latent space. Therefore, ensuring the stealthiness of the attack method from the latent space perspective is crucial. We visualized the embedding features of BADFSS using

Table 3. Defense evaluation results.

Pre-training Dataset	Downstream Dataset	Neural Cleanse [30] Anomaly Index	DECREE [6] $\mathcal{PL}^1$-Norm
CIFAR-10	STL-10	1.11	0.26
	GTSRB	1.16	0.37
	SVHN	1.32	0.19
STL-10	CIFAR-10	0.98	0.23
	GTSRB	1.22	0.31
	SVHN	1.37	0.25

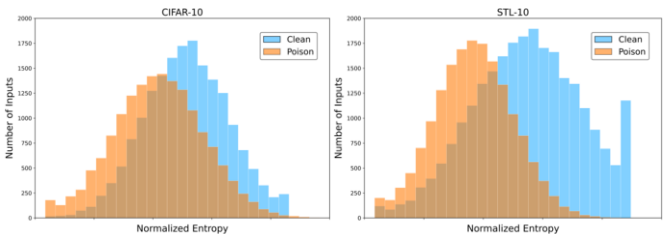

Figure 6. Experimental results of STRIP.

PCA, as shown in Figure 4. In the left diagram, we observe that the features of BADFSS form two distinct clusters, indicating that they can be easily detected in the latent space by any clustering algorithm. In contrast, in our IPBA, the feature representations of poisoned samples are intermingled with those of benign samples, forming a single cluster. This demonstrates that IPBA achieves optimal stealthiness in the latent space, breaking the assumption of latent separation, and effectively evading backdoor defenses.

5.4 Robustness Evaluation (RQ3)

To evaluate the robustness of IPBA against existing backdoor defenses, we implemented representative backdoor defense methods (i.e., DECREE [6], STRIP [7], Neural Cleanse [30], and GradCAM [26]) and assessed IPBA's resistance to these defenses. Additional defense results are provided in the appendix [31].

Resistance to DECREE. DECREE [6] detects backdoor attacks in pre-trained encoders by reversing the trigger. If the reversed trigger has a $\mathcal{PL}^1$-Norm smaller than 0.1, the encoder is considered compromised. As shown in Table 3, all backdoor triggers generated by IPBA have a $\mathcal{PL}^1$-Norm above 0.1, successfully evading detection by DECREE.

Resistance to STRIP. STRIP [7] is a sample-based backdoor detection method. It assumes that a backdoored model's predictions exhibit stability on malicious samples, detecting such samples by computing entropy after overlaying random samples. Figure 6 shows that STRIP fails to establish an effective threshold to distinguish benign from malicious samples, allowing our attack to bypass detection.

Resistance to Neural Cleanse. Neural Cleanse (NC) [30] is a defense method against trigger generation that measures the deviation of the reconstructed trigger by calculating the anomaly index, marking models with an anomaly index greater than 2 as backdoored. Since NC is designed specifically for classifiers and cannot be directly applied to image encoders, we use NC to identify backdoors in downstream classifiers. Experimental results (see Table 3) show that the anomaly index of IPBA is below 2 across all datasets, successfully evading detection by NC.

Resistance to GradCAM. GradCAM [26] generates heatmaps to show the contribution of each pixel in the model's prediction, with backdoored models typically exhibiting abnormal heatmaps. Figure 7 shows the visualization heatmaps of a clean model and a backdoored model attacked by IPBA. From this figure, we can find that the heatmaps of these models are very similar, and IPBA is capable of resisting defense methods based on GradCAM.

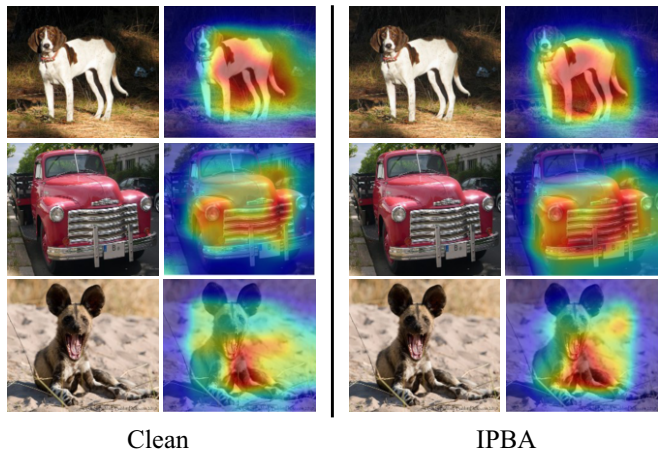

Figure 7. GradCAM visualization results for both clean and backdoored models.

Table 4. Performance of Ablation Studies.

$\mathcal{L}_{ste}$	$\mathcal{L}_{dis}$	$\mathcal{L}_{align}$	BA↑	ASR↑	SSIM↑	PSNR↑	LPIPS↓
✓	✓	✓	87.13	99.94	0.9898	35.38	0.0026
✓		✓	90.35	42.32	0.9864	32.08	0.0048
✓	✓		62.25	9.68	0.9802	30.16	0.0046
	✓	✓	84.66	95.27	0.1937	6.79	0.6534

5.5 Ablation Study

In this section, we demonstrate the effects of each important component in Section 4.3 by ablating them respectively. As shown in Table 4, after ablating each component, the ASR of our method decreases across the target downstream datasets. Specifically, when ablating $\mathcal{L}_{dis}$ and $\mathcal{L}_{align}$ sequentially, the ASR of IPBA decreases from 99.94% to 42.32% and 9.68%, respectively, which is consistent with our previous observations. When ablating $\mathcal{L}_{ste}$, the SSIM of IPBA decreases from 0.9898 to 0.1937, the PSNR drops from 35.38 to 6.79, and the LPIPS increases from 0.0026 to 0.6534. These changes clearly indicate that the absence of key components significantly weakens the effectiveness of our backdoor attack method.

6 Conclusion

This paper introduces IPBA, an imperceptible and effective backdoor attack method for FSSL. IPBA decouples the feature distributions of backdoor and augmented samples and incorporates Sliced Wasserstein Distance to effectively address the challenges and limitations of existing imperceptible triggers in FSSL. Results show that IPBA outperforms baseline methods and is effective under different settings. We further explore potential countermeasures against our attack and find that existing defense mechanisms are insufficient to mitigate IPBA, indicating the need for specifically designed defense strategies to alleviate backdoor attacks in FSSL.

Acknowledgements

This work was supported by grants 24KJB520042 (Jiangsu), 2025YSZ-017 (Yangzhou), 2023SGJ014 (Hefei), COGOS-2023HE01 (iFLYTEK), Y202352288 (Zhejiang), and 2023AY11057 (Jiaxing), as well as by resources from Microsoft Azure and the NSF-supported Chameleon testbed.

References

[1] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, pages 2938–2948. PMLR, 2020.

[2] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

[3] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[4] A. Coates, A. Ng, and H. Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

[5] 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*, 2021.

[6] S. Feng, G. Tao, S. Cheng, G. Shen, X. Xu, Y. Liu, K. Zhang, S. Ma, and X. Zhang. Detecting backdoors in pre-trained encoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16352–16362, 2023.

[7] 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*, pages 113–125. ACM, 2019.

[8] J.-B. Grill, F. Strub, F. Altch'e, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[9] C. He, Z. Yang, E. Mushtaq, S. Lee, M. Soltanolkotabi, and S. Avestimehr. Ssfl: Tackling label deficiency in federated learning via personalized self-supervision. *arXiv preprint arXiv:2110.02470*, 2021.

[10] 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*, pages 770–778, 2016.

[11] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

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

[13] J. Jia, Y. Liu, and N. Z. Gong. Badencoder: Backdoor attacks to pre-trained encoders in self-supervised learning. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 2043–2059. IEEE, 2022.

[14] S. Kolouri, K. Nadjahi, U. Simsekli, R. Badeau, and G. Rohde. Generalized sliced wasserstein distances. *Advances in neural information processing systems*, 32, 2019.

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

[16] C. Li, R. Pang, Z. Xi, T. Du, S. Ji, Y. Yao, and T. Wang. An embarrassingly simple backdoor attack on self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4367–4378, 2023.

[17] H. Liu, J. Jia, and N. Z. Gong. {PoisonedEncoder}: Poisoning the unlabeled pre-training data in contrastive learning. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 3629–3645, 2022.

[18] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A. Y. Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 4. Granada, 2011.

[19] A. Nguyen and A. Tran. Wanet–imperceptible warping-based backdoor attack. In *International Conference on Learning Representations*, 2021.

[20] T. D. Nguyen, T. A. Nguyen, A. Tran, K. D. Doan, and K.-S. Wong. Iba: Towards irreversible backdoor attacks in federated learning. *Advances in Neural Information Processing Systems*, 36:66364–66376, 2023.

[21] Y. Qian, S. Wu, K. Wei, M. Ding, D. Xiao, T. Xiang, C. Ma, and S. Guo. Eminspector: Combating backdoor attacks in federated self-supervised learning through embedding inspection. *arXiv preprint arXiv:2405.13080*, 2024.

[22] Y. A. U. Rehman, Y. Gao, P. P. B. De Gusmão, M. Alibeigi, J. Shen, and N. D. Lane. L-dawa: Layer-wise divergence aware weight aggregation in federated self-supervised visual representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16464–16473, 2023.

[23] 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*, pages 234–241. Springer, 2015.

[24] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.

[25] A. Saha, A. Tejankar, S. A. Koohpayegani, and H. Pirsiavash. Backdoor attacks on self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13337–13346, 2022.

[26] 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*, pages 618–626, 2017.

[27] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32:323–332, 2012.

[28] G. Tao, Z. Wang, S. Feng, G. Shen, S. Ma, and X. Zhang. Distribution preserving backdoor attack in self-supervised learning. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 2029–2047. IEEE, 2024.

[29] B. Van Berlo, A. Saeed, and T. Ozcelebi. Towards federated unsupervised representation learning. In *Proceedings of the third ACM international workshop on edge systems, analytics and networking*, pages 31–36, 2020.

[30] 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)*, pages 707–723. IEEE, 2019.

[31] J. Wang, Y. Song, Z. Zhao, J. Zhang, Q. Wu, J. Zhu, and D. Zhao. Ipba: Imperceptible perturbation backdoor attack in federated self-supervised learning. *arXiv preprint arXiv:2508.08031*, 2025. Full version of this paper.

[32] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[33] S. Wu, C. Ma, K. Wei, M. Ding, J. Yang, and Y. Qian. Towards efficient backdoor attacks against federated self-supervised learning as a service through intra-union aggregation. In *International Conference on Service Science*, pages 122–135. Springer, 2024.

[34] F. Zhang, K. Kuang, L. Chen, Z. You, T. Shen, J. Xiao, Y. Zhang, C. Wu, F. Wu, Y. Zhuang, et al. Federated unsupervised representation learning. *Frontiers of Information Technology & Electronic Engineering*, 24(8): 1181–1193, 2023.

[35] H. Zhang, Z. Wang, T. Han, M. Jin, C. Zhan, M. Du, H. Wang, and S. Ma. Towards imperceptible backdoor attack in self-supervised learning. *arXiv preprint arXiv:2405.14672*, 2024.

[36] J. Zhang, H. Liu, J. Jia, and N. Z. Gong. Data poisoning based backdoor attacks to contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24357–24366, 2024.

[37] J. Zhang, C. Zhu, X. S. Di Wu, J. Yong, and G. Long. Badfss: Backdoor attacks on federated self-supervised learning. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI-24)*, pages 548–558, 2024.

[38] 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*, pages 586–595, 2018.

[39] W. Zhuang, Y. Wen, and S. Zhang. Divergence-aware federated self-supervised learning. In *International Conference on Learning Representations*, 2022.