TTA-FedDG: Leveraging Test-Time Adaptation to Address Federated Domain Generalization

Haoyuan Liang , Xinyu Zhang , Shilei Cao , Guowen Li , Juepeng Zheng*

School of Artificial Intelligence , Sun Yat-sen University , China
{lianghy68 , zhangxy869 , caoshlei , ligw8}@mail2.sysu.edu.cn , zhengjp8@mail.sysu.edu.cn

Abstract

In recent years, Federated Domain Generalization (FedDG) has succeeded in generalizing to unknown clients (domains). However, current methods only utilize training data, and when there is a significant difference between the unknown client and source client domains (**domain shift**), these methods cannot ensure model performance. This limitation appears to have caused research in FedDG to reach a bottleneck. On the other hand, test data is a resource that can help models adapt while previous FedDG approaches have not taken this into account. In this paper, we introduce a new framework TTA-FedDG to address the FedDG problem, which leverages test-time adaptation (TTA) to adapt across different domains, thereby enhancing the generalization of the model. We propose the method Federated domain generalization based on select Strong Pseudo Label (FedSPL), which combines fast feature matching and knowledge distillation. Our method consists of two parts. **Firstly**, we use fast feature reordering for feature mixing during local updates on the client side, improving the robustness of the global model and enhancing its generalization ability to mitigate domain shift. **Secondly**, we employ a teacher-student model with contrastive learning and label selection during the testing phase, enabling the global model to better adapt to the distribution of the target client, thereby alleviating domain shift. Extensive experiments have demonstrated the effectiveness of FedSPL in handling domain shift, outperforming existing FedDG methods across multiple datasets and model architectures.

Introduction

As deep learning models(Zhang et al. 2024a,b) continue to evolve, more data is required to train larger models, such as ChatGPT. However, some clients consider their data private, making them reluctant to share it for training, resulting in data silos. Federated Learning (FL) (McMahan et al. 2017) has emerged as a solution to address data privacy and data silo issues, garnering significant attention in recent years. However, in the traditional FL paradigm, each client trains a local model using their data and then aggregates these models into a global model to accommodate the participating clients. Various approaches, such as adaptive weights (Yu, Bagdasaryan, and Shmatikov 2020) and personalized federated learning (Shamsian et al. 2021), have been proposed to improve the performance of global models on local clients. However, as FL becomes more widespread, it attracts new clients who want to use the global model on their dataset. Traditional Federated Learning faces challenges in adapting effectively to new clients, particularly when the data distribution of these new clients differs significantly from that of the training clients. To address this, Liu et al. (2021) proposed the Federated Domain Generalization (FedDG) paradigm and the ELCFS method to enhance the generalization of the global model to new clients. However, in ELCFS, sharing distribution information during client local training can lead to privacy leaks. Consequently, much of the current research has shifted toward leveraging local models during the aggregation phase, leading to the development of adaptive weighting methods such as FedDG-GA (Zhang et al. 2023) within the

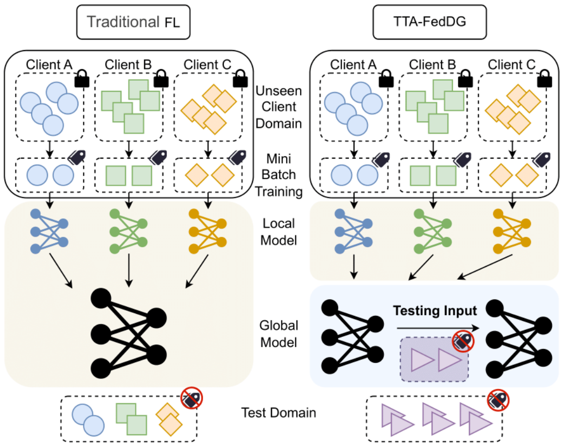

Figure 1: This illustration highlights the two major differences between FL and TTA-FedDG. First, the clients served by the global model differ: FL emphasizes the applicability of the model to known clients, whereas TTA-FedDG focuses on its generalization to unknown clients. Second, in TTA-FedDG, we use unlabeled test data to adapt the global model, which is not the case in traditional FL.

*Corresponding Author.

FedDG paradigm.

Although these adaptive weighting methods can improve the generalization of the global model to unknown domains to some extent, they may underperform in certain domains compared to FedAvg due to the lack of knowledge about the distribution of these unknown domains. This over-reliance on training data seems to have caused traditional FedDG methods to reach a bottleneck.

A natural question arises: how can the model utilize unknown test data to achieve self-adaptation? We believe that test data can be treated as unlabeled training data, thus enabling unsupervised learning during testing. Therefore, we propose a new framework to address the FedDG problem: Test-time Adaptation Federated Domain Generalization (TTA-FedDG). As shown in Fig.1, the differences between FL and TTA-FedDG are illustrated. Then, we propose the method of Federated domain generalization based on select Strong Pseudo Label (FedSPL) which Combines Feature ordering in training and Knowledge Distillation in test time to overcome domain shift for unknown clients. Feature mixing has already shown significant improvements in domain generalization (DG), as seen in approaches like MixStyle (Zhou et al. 2021). Feature mixing introduces a broader range of feature combinations during client-side training, thereby improving the model's generalization ability. However, feature mixing methods based on AdaIN (Huang and Belongie 2017) typically only match limited-order statistics, such as first and second-order moments, which can significantly affect the quality of the mixed features. Therefore, firstly, we apply a matching and mixing approach based on feature ranking, enabling FedDG to match all-order statistics during client training. Furthermore, we are the first to use TTA methods in FedDG. Secondly, we propose an improved teacher-student model to enable the global model better to learn the distribution of new client data during testing, thereby enhancing the performance of the model in unknown domains. The framework of our method FedSPL, is illustrated in Fig.2.

The traditional teacher-student model overly relies on pseudo-labels generated by the teacher model, which hampers the ability of the model to learn the new client data distribution effectively. Therefore, selecting high-quality pseudo-labels is a significant challenge. Previous works have used methods like sorting by the maximum value of the classifier or using KL divergence (Hershey and Olsen 2007) to determine the confidence level of the pseudo-labels. In contrast, we design a variable t defined as the difference between the highest and second-highest values of the classifier, which better reflects the confidence of the model in the classification result. Based on the size of this variable, we divide the samples and pseudo-labels into high-quality and low-quality groups. The student model is trained sequentially with these groups, using different weights and loss functions for each. Additionally, to address the memory issue during the training of the student model, we incorporate an optimized contrastive loss (Wang and Liu 2021). Since the size of image pixels affects brightness and their relationships affect contours, we normalize the input features before performing contrastive learning. Our main contributions are summarized as follows:

- We propose a new framework to address the FedDG problem: Test-time Adaptation Federated Domain Generalization (TTA-FedDG), which leverages test-time adaptation (TTA) techniques. TTA can learn the domain distribution of new clients during the testing phase, thereby enhancing the generalization of the model.
- We propose a method called Federated domain generalization based on select Strong Pseudo Label (FedSPL). This method uses feature mixing based on feature ranking to enhance the generalization ability of client models and applies an improved teacher-student model during the testing phase to adapt the model to the distribution of new client domains.
- We conducted extensive experiments to demonstrate the state-of-the-art performance and effectiveness of our TTA-FedDG framework in handling FedDG tasks.

Related Work

Federated Domain Generalization (FedDG)

Domain Generalization (DG) aims to address the issue of model overfitting due to limited domain-specific data (Li et al. 2019). DG allows for the use of data from multiple domains during training, enhancing the generalization ability of the model without the need for domain-specific knowledge. Current research in DG can be categorized into three main approaches. Techniques such as style augmentation (Jackson et al. 2019) enhance data diversity through feature mixing, preventing overfitting. Methods like JiGen (Carlucci et al. 2019; Du et al. 2020) aim to capture the characteristics of each domain and promote domain knowledge sharing to boost model generalization. However, joint domain generalization faces challenges due to privacy concerns, as data from different domains cannot be pooled together for training (Sun, Chong, and Ochiai 2023). This limitation hinders the applicability of many existing DG methods. Federated domain generalization is an emerging field where each domain trains a local model which is then aggregated, considering the generalization ability of the global model on target clients (or domains). Despite recent progress, research in this area is still limited. For example, ELCFS (Liu et al. 2021) generalizes the model by sharing spectra across domains, yet this approach risks privacy leaks. Adaptive weighting methods (Zhang et al. 2023) have also been proposed, but these still face significant challenges.

However, current methods (Wei and Han 2024; Zhou et al. 2024) only utilize training data, overlooking test data that valuable knowledge can provide to the model.

Test-Time Adaptation(TTA)

Test-Time Adaptation (TTA) (Karim et al. 2023; Cao et al. 2024) is a technique that improves model performance and generalization by fine-tuning or adapting to new data during the inference phase. Hypothesis Transfer Learning (HTL) (Kuzborskij and Orabona 2013) is one of the methods TTA uses to address domain generalization. It leverages prior knowledge (hypotheses) learned from related tasks to aid and accelerate the learning process of a new task, particularly when labeled data for the new task is scarce. However,

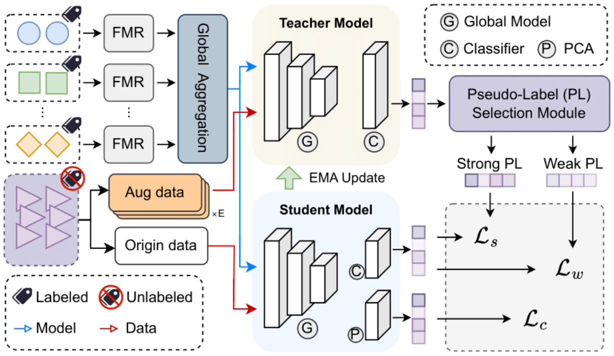

Figure 2: This figure illustrates the framework of our method, FedSPL. We employ the Model Based on Strong pseudo-Labels (MBSL) by incorporating label selection, hierarchical training, and contrastive learning into the teacher-student model training process. This approach mitigates domain shift and memory issues during training. Additionally, since TTA training can introduce instability, we enhance the stability of the teacher-student model by incorporating Feature Mixing and feature Reordering (FMR) during local training.

in TTA-FedDG, the global model does not have access to the test set labels, making this type of method unsuitable for TTA-FedDG. TTA has been employed in Personalized Federated Learning and has shown excellent results, such as in FedTHE (Jiang and Lin 2022) and ATP (Bao et al. 2024).

However, TTA is more necessary when dealing with unknown clients. Although ATP (Bao et al. 2024) has involved generalization to unknown clients in part experiments, ATP is designed to address feature shift and label shift in TTA-PFL. Therefore, it has not yet been proposed that the TTA-FedDG framework be used to address the FedDG problem.

Method

Preliminaries

In this paper, we propose a new framework TTA-FedDG, which leverages test-time adaptation to address federated domain generalization (FedDG). In FedDG, this is a more stringent setting compared to heterogeneous FL, as it requires each client to have its domain distribution, with each client domain being distinct from the others. In other words, the data distribution for each client k follows the domain D_k. Additionally, the source domain distribution $\{D_k\}_{k=1}^{N}$ and the target domain distribution $\{T_i\}_{i=1}^{M}$ are not the same, which is another key difference between FL and FedDG.In previous work, the focus has predominantly been on learning from data samples $\{(x_1^{D_k}, y_1^{D_k}), \cdots, (x_{n_k}^{D_k}, y_{n_k}^{D_k})\}$ within each source client k where n_k represents the size of the dataset for source client k.Although test samples $\{x_1^{T_i}, \cdots, x_{m_i}^{T_i}\}$ *lack labels*, they are not unusable where m_i represents the size of the dataset for test client i.Then, we will explain how to adjust the model using the aggregated global model and target data.

Feature Mixing Based on Feature Reordering(FMR)

Most TTA methods exhibit significant fluctuations, indicating high instability. To address this, we incorporated image augmentation techniques during the TTA adaptation process to enhance model robustness. Additionally, we implemented Feature Mixing Based on Feature Reordering (FMR) to further strengthen the robustness of local models $f(w_{l_k};\cdot)$. The global model $f(w_g;\cdot)$ is then obtained by linearly weighting the local models (as non-linear weighting with FMBFR might not yield optimal results). This approach enhances the robustness of $f(w_t;\cdot)$, $f(w_g;\cdot)$ and $f(w_s;\cdot)$.

Our method FMR is inspired by (Zhang et al. 2022), where we implement fast histogram matching to achieve feature mixing without losing higher-order statistics. This approach ensures that the samples used for training differ each time, as the features are randomly shuffled. This effectively increases the diversity of the samples, thereby enhancing the robustness of the $f(w_{l_k};\cdot)$.

As shown in Fig.2, we illustrate the role of FMR in local training. Eq.1 demonstrates the state before shuffling, while Eq.2 shows the state after shuffling.

$$B_{original} = [F(x_{B_1}^{D_k}), \cdots, F(x_{B_q}^{D_k}), \cdots, F(x_{B_n}^{D_k})] \tag{1}$$

$$B_{shuffle} = [F(x_{B_i}^{D_k}), \cdots, F(x_{B_p}^{D_k}), \cdots, F(x_{B_j}^{D_k})] \tag{2}$$

For $F(x_{B_p}^{D_k})$ and $F(x_{B_q}^{D_k})$, we sort them and assign the indices of $F(x_{B_q}^{D_k})$ to $F(x_{B_p}^{D_k})$ to achieve feature mixing. This mixing strategy, when applied to images, can greatly preserve the contours of $F(x_{B_q}^{D_k})$ while retaining the colors of $F(x_{B_p}^{D_k})$. The blending method is detailed in Eq.3 :

$$\begin{aligned} &F^{aug_p}(x_{B_q}^{D_k}) = Index_q(Sort(F(x_{B_p}^{D_k}))) \\ &s.t.\ Index_q = Index(F(x_{B_q}^{D_k})) \end{aligned} \tag{3}$$

The size relationship between image pixels is positively correlated with feature values. Therefore, to preserve the contours of $F(x_{B_q}^{D_k})$, we must ensure that the order of $F(x_{B_q}^{D_k})$ feature values remains unchanged while using $F(x_{B_p}^{D_k})$ feature values for rendering. The final batch that enters the network is shown in Eq.4:

$$B_{final} = [F^{aug_i}(x_{B_1}^{D_k}) \cdots F^{aug_p}(x_{B_q}^{D_k}) \cdots F^{aug_j}(x_{B_n}^{D_k})] \tag{4}$$

It is worth mentioning that our mixing results are not identical each time because the shuffle order is random. This means that the batches entering the local model are different each time. By leveraging this randomness and the feature mixing method, we enhance the diversity of the samples, thereby increasing the robustness of the model.

Model Based on Strong pseudo-Labels(MBSL)

Teacher Student Model is divided into two sequential parts: the first part generates pseudo-labels through the teacher model, and the second part trains the student model based on these generated pseudo-labels, thereby updating the student model parameters w_s.After the server aggregates the local client models to obtain the global model parameters w_g, the global model is passed to both the teacher model and the student model. Therefore, at the start of TTA, *i.e.*,$w_s = w_t = w_g$.

After the student model updates w_s, the parameters of the teacher model w_t are updated through the Exponential Moving Average (EMA) (Hansun 2013). The update method for EMA is as follows Eq.5:

$$w_t^{j+1} = \rho w_t^j + (1-\rho) w_s^{j+1} \tag{5}$$

where j is the updating number, and ρ represents the adjustment coefficient, a hyperparameter that indicates the level of trust in the teacher and student models.

To enhance the generalization ability of the teacher model, we expanded the target original data by a factor of E (=10) through image augmentation. We then used the augmented target data to train the teacher model, while the original target data was used to train the student model. The pseudo-labels generated by the teacher model are computed using Eq.6:

$$\begin{aligned} & y_{\mathrm{t}}^i = \arg\max \bar{\boldsymbol{f}}(w_t; x_i^{T_i}) = \arg\max \frac{1}{E} \sum_{e=1}^{E} f_e(w_t; x_i^{T_i}) \\ & s.t.\ \boldsymbol{f}(w_t; x_i^{T_i}) = \{f_1(w_t; x_i^{T_i}), \cdots, f_e(w_t; x_i^{T_i})\} \end{aligned} \tag{6}$$

where we incorporate image augmentation as an integral part of the network. $f_e(\cdot)$ represents the input to the model after e-level image augmentation and $\bar{\boldsymbol{f}}(\cdot)$ represents the mean of these models.

Selecting Strong Pseudo-labels is key to improving global model performance. The traditional Teacher Student Model generates pseudo-labels in one go based on the teacher model, which can result in pseudo-labels being overly similar to the source domain label distribution and this makes it difficult to effectively address domain shift. This does not fully leverage the potential of pseudo-labels. We proposed a new pseudo-label learning strategy based on the Curriculum teacher student model(Karim et al. 2023). We classify the target dataset into two categories based on the reliability of the pseudo-labels: **Strong** and **Weak**. Although there have been some previous classification standards (Qiao and Peng 2021) based on traditional classifier statistics, such as mean entropy and variance, these methods struggle when the probability given by the classifier is dispersed and concentrated. For example, these methods become difficult to apply effectively in a classification task with four classes, where the predicted probabilities are 0.5, 0.4, 0.05, and 0.05. Therefore, we propose a new standard for classifying pseudo-labels.

Our insight is that even if the maximum predicted probability is relatively small, it does not necessarily indicate a lack of confidence in the prediction. It may simply reflect sufficient confidence in excluding other classes. Therefore, we propose the confidence separation score (**CSS**), which we define as the difference between the maximum predicted probability and the second-highest value. The details are as follows Eq.7:

$$CSS_m = max(f(w_t; x_m^{T_i})) - 2nd\_max(f(w_t; x_m^{T_i})) \tag{7}$$

where *max* represents the maximum value, *2nd_max* represents the second maximum value, and $f(\cdot)$ denotes the predicted probability output of model.

We determine whether a label is **strong** or **weak** based on whether it exceeds a threshold δ, as shown in Eq.8:

$$\begin{aligned} & \tau^m = \begin{cases} 1, & if\, CSS_m \geq \delta \\ 0, & \text{otherwise.} \end{cases} \\ & s.t.\ \delta = \frac{1}{m_i} \sum_{m=1}^{m_i} CSS_m\ ,\ m \in (1, m_i) \end{aligned} \tag{8}$$

when $\tau = 1$, we consider it a **strong** pseudo-label, and when $\tau = 0$, it is a **weak** pseudo-label. We represent the threshold τ using the mean of the CSS. δ varies with different target clients, making it no longer a fixed hyperparameter, which allows the pseudo-label selection process to be adaptive. After labeling each sample, a batch will contain two types.When $\tau^i = 1$, $\mathbb{B}_S^{T_i} = \{(x_i^{T_i}, y_{\mathrm{t}}^i)\}_{i=1}^B$ and when $\tau^i = 0$, $\mathbb{B}_W^{T_i} = \{(x_i^{T_i}, y_{\mathrm{t}}^i)\}_{i=1}^B$. where y_{t}^i is the pseudo-labels generated by the teacher model for the sample $x_i^{T_i}$.

Design Different Loss Functions. We apply different strategies for **strong** and **weak** labels. Since we consider $\mathbb{B}_S^{T_i}$ to be highly confident, we treat them as true labels and calculate the cross-entropy (*CE*) loss, as shown in Eq.9, to update the parameters of the student model.

$$\mathcal{L}_s = -\frac{1}{|\mathbb{B}_S^{T_i}|} \sum_{i=1}^{|\mathbb{B}_S^{T_i}|} y_{\mathrm{t}}^i \cdot \log f(w_t; x_i^{T_i})) \tag{9}$$

However, due to the differences in sample types, the output of classifier results may vary. If we only use the $\mathbb{B}_S^{T_i}$, the model might only learn to classify easier classes, while more challenging classes could fall into the $\mathbb{B}_W^{T_i}$, which would not improve the performance of the model on those difficult-to-classify classes. To fully leverage the target data, we also include $\mathbb{B}_W^{T_i}$ in the learning process, but assign them a smaller gradient value during gradient descent. To ensure that more emphasis is placed on learning from the strong class initially, we set the starting parameter value from 0 to 0.005. For $\mathbb{B}_W^{T_i}$, we compare the classifiers of the teacher model and the student model, aiming for them to produce consistent prediction results. The loss (Zhou et al. 2003; Karim et al. 2023) we apply is shown in Eq.10:

$$\mathcal{L}_w = \frac{1}{|\mathbb{B}_W^{T_i}|} \sum_{i=1}^{|\mathbb{B}_W^{T_i}|} \|\bar{\boldsymbol{f}}(w_t; x_m^{T_i}) - f(w_s; x_m^{T_i})\|^2 \tag{10}$$

However, the pseudo-labels used for training the student model are generated by the teacher model using the same data, which can lead to a memory issue. When the label quality is poor, including the performance of strong labels, the memory effect can cause the performance of the model to stagnate during the adaptation phase, making it difficult to achieve improvement. Therefore, we introduce contrastive learning to help the model distinguish between similar and dissimilar features, thereby alleviating the memory issue and continuously improving its performance during the test phase. For any test sample $x_m^{T_i}$, we select two augmented versions from the teacher model, version 1 $aug_1(x_m^{T_i})$ and version e $aug_e(x_m^{T_i})$, to perform contrastive learning (*CL*), thereby enhancing the robustness of the model.

Next, we perform feature extraction $F(\cdot)$ on $aug_1(x_m^{T_i})$ and $aug_e(x_m^{T_i})$ obtaining features $F(aug_1(x_m^{T_i}))$ and $F(aug_e(x_m^{T_i}))$. Since sample augmentation may introduce noise, we apply Principal Component Analysis (PCA) to the extracted features to isolate the principal features $\boldsymbol{p}_m = PCA(F(aug_1(x_m^{T_i})))$ and $\boldsymbol{q}_m = PCA(F(aug_e(x_m^{T_i})))$. We then conduct *CL* on $\boldsymbol{q}_m$ and $\boldsymbol{p}_m$. Here, both the $aug_e(\cdot)$ and the $F(\cdot)$ are components of the model $f_e(\cdot)$.

We believe that traditional *CL*, which directly computes similarity between features, can adapt to various tasks. However, for image data, pixel values primarily reflect brightness, while the size relationship between adjacent pixels characterizes the contour of the image. Therefore, leveraging this characteristic of images, we normalize the principal features after image processing before computing similarity. This approach significantly mitigates dissimilarities caused by variations in image brightness. The *CL* loss we use is shown in Eq.11:

$$\begin{aligned} &\mathcal{L}_c = -\frac{1}{B}\sum_{m=1}^{B} \log \frac{\exp(sim(P_m, Q_m)/T)}{\sum_{b=1}^{B} \phi_b}, \\ &s.t. \quad P_m = \boldsymbol{p_m} - \bar{\boldsymbol{p}}_{\boldsymbol{m}}, \;\; Q_m = \boldsymbol{q_m} - \bar{\boldsymbol{q}}_{\boldsymbol{m}}, \\ &\phi_b = \exp(sim(P_b, Q_b)/T) + 1_{b \neq m} \exp(sim(P_m, P_b)/T) \end{aligned} \tag{11}$$

where T is a temperature constant. $sim(\cdot)$ is Cosine Similarity. Through hierarchical learning of pseudo-labels and leveraging them, we obtain the total loss as shown in Eq.12:

$$\mathcal{L}_t = (1 - \alpha_j - \beta_j)\mathcal{L}_s + \alpha_j \mathcal{L}_w + \beta_j \mathcal{L}_c \tag{12}$$

Since we prioritize learning from **strong** samples, the initial value of α is set to $\alpha_0 = 1$, with an upper bound of $\eta = 0.001$. Although we aim to learn from all samples, we remain cautious about **weak** samples, and thus, their contribution to the loss is minimal. Therefore, the update strategy for α is defined as shown in Eq. 13 below:

$$\alpha_j = \alpha_{j-1}(1 - \delta) \tag{13}$$

Similarly, CL can help the model better learn similar class features and distinguish between different classes in the early stages. However, as iterations progress, its impact becomes less significant. Therefore, we apply exponential decay to the coefficient β, with the update defined as shown in Eq. 14:

$$\beta^j = \beta^{j-1} e^{-\gamma} \tag{14}$$

Here, we set the initial values β_0 as 0.4 and set γ to 1e-3.

The pseudocode of our method is in Algorithm 1.

Experiments

Setup

In this section, we briefly introduce the experimental setup. Unless otherwise specified, all experiments will be conducted under the following conditions. The default settings are as follows:

Algorithm 1: Federated domain generalization base on select Strong Pseudo Label(FedSPL)

```
Initial global model $w = w_0$, $k$ source domain client $D = \{D_1, D_2, ..., D_k\}$, the weights $\mathbf{q} = \{\frac{n_1}{N}, \frac{n_2}{N}, .., \frac{n_k}{N}\}$,datasets size of domain client $e$ is $n_e$, $N = n_1 + n_2 ... + n_k$. (Hyper-parameters:local epoch $T$,total aggregation round $R$)
Final global model $w_R$
Server:
    Deploy the global model $w_0$ to each domain client to obtain each client model:$w_0^k = w_0$
    for each round $i = 1, 2, ..., R$ do
        for each domain client $e = 1, 2, ..., k$ in parallel do
            $w_{i+1}^e \leftarrow$ Client ( $e$ , $w_i$ )
        $w_{i+1} \leftarrow \sum_{e=1}^{k} q^e w_{i+1}^e$
        Deploy $w_{i+1}$ to all domain clients.
Client:
    for each local epoch $t = 1, 2, ..., T$ do
        for Batch $b = 1, 2, ..., B$ do
            B$_{final} = FMR(B_{original})$
            $w_{i+(t+1)} \leftarrow w_{i+(t)} - \eta \nabla \mathcal{L}(w_{i+(t)}; F(x_{B_q}^{D_k}))$
    return $w_{i+1} = w_{i+(T)}$ to Server
Test:
    The initial setting:$w_t = w_s = w_R$
    for each epoch $j = 1, 2, ..., T'$ do
        $w_s^{j+1} = MBSL(w_s^j)$
        $w_t^{j+1} = \rho w_t^j + (1 - \rho) w_s^{j+1}$
    return $w_t$ to target client
```

Dataset. We evaluate our proposed method on three widely used domain benchmarks. (*i*) The **PACS** (Li et al. 2017) dataset contains four distinct domains (*Photos, Art Paintings, Cartoons, and Sketches*), with a total of 9,991 images. Each domain shares the same 7-class label space, despite the variations in image styles. To evaluate the generalization capability of our model, we followed the standard split scheme for training and validation, and we conducted extensive ablation experiments on this dataset. (*ii*) The **Office-Home** (Venkateswara et al. 2017) dataset contains approximately 15,500 images across 65 categories from four domains (*Art, Clipart, Product, and Real-World*), offering a diverse range of categories that better test the robustness of our method. (*iii*) The **Digit-5**(Wei and Han 2024) dataset is designed for digit recognition and includes images from five different domains: *MNIST (mt), SVHN (sv), USPS (up), Synth (syn), and MNIST-M (mm)*. The multi-domain nature of this dataset allows us to test the generalization.

Comparing Methods. We selected several representative methods from the TTA and FL fields for comparison, with FedAvg (McMahan et al. 2017) serving as the baseline method. FedAvg is known for its stable performance with unknown target clients. In our comparisons, we selected classic algorithms from FL, FedDG, and TTA, with ATP and GA serving as the *state-of-the-art* methods for TTA-FedDG and FedDG, respectively.Among them, FedProx (Li et al. 2020), Fed-

Method	TTA	PACS					OfficeHome				
		Photo	Art	Cartoon	Sketch	Avg	Product	Art	Clipart	Real	Avg
FedAvg	×	92.77	77.29	77.97	81.03	82.26	72.72	57.60	52.28	73.88	64.12
FedProx	×	93.15	77.72	77.73	80.77	82.34	73.37	58.76	52.67	73.88	64.67
FedCSA	×	91.88	77.00	76.79	80.84	81.63	72.96	57.58	53.99	73.98	64.63
FedNova	×	94.03	79.93	76.39	79.26	82.40	73.72	58.81	49.89	73.33	63.94
AM	×	93.29	80.86	77.62	81.05	83.20	73.24	58.76	51.87	73.84	64.42
RSC	×	92.67	77.98	77.80	82.90	82.91	73.26	57.44	50.31	73.42	63.61
FedSAM	×	91.20	74.45	77.77	**83.35**	81.69	73.58	55.34	54.75	73.74	64.35
HarmoFL	×	90.99	74.51	77.43	81.73	81.16	73.89	57.44	53.42	74.95	64.93
Scaffold	×	92.50	78.09	77.23	80.67	82.12	72.16	59.00	52.78	73.22	64.29
ELCFS	×	93.31	82.23	74.77	82.27	83.24	71.11	57.85	54.93	73.71	62.27
GA	×	93.97	81.28	76.73	82.57	83.64	73.39	58.57	54.39	74.73	65.27
DSBN	✓	96.26	82.83	80.99	77.50	84.25	73.34	56.49	53.64	73.03	64.13
Tent	✓	96.56	85.94	83.06	81.39	86.83	74.63	57.95	56.48	74.67	65.92
ATP	✓	95.52	82.96	79.46	82.28	85.04	75.31	59.97	**56.72**	**75.21**	66.80
FedSPL (ours)	✓	**98.14**	**89.38**	**84.72**	82.84	**88.77**	**76.56**	**60.86**	55.88	74.81	**67.03**

Table 1: Results (%) of PACS on ResNet18 and Results (%) of OfficeHome on ResNet50 (Best in **bold**)

CSA (Ma et al. 2021), FedNova (Wang et al. 2020b) ,AM (Qu et al. 2022), RSC (Huang et al. 2020), FedSAM (Qu et al. 2022), HarmoFL (Jiang, Wang, and Dou 2022), Scaffold(Karimireddy et al. 2020) and GA (Zhang et al. 2023) are algorithms used to optimize FedDG, while DSBN (Jiang et al. 2022), Tent (Wang et al. 2020a), and ATP (Bao et al. 2024) are methods applied for TTA.

Implementation Details. For local model training across the PACS, OfficeHome, and Digit-5 datasets, we utilize architectures ResNet18 and ResNet50, as detailed by (He et al. 2016), which are pre-trained on the ImageNet database (Deng et al. 2009). We adopt a leave-one-domain-out evaluation method for all benchmarks, where one domain is reserved for testing and the remaining domains are used for training and validation purposes. To ensure consistency and fairness in our experiments, we standardize the batch size and learning rate at 128 and 0.2, respectively, during local training. Furthermore, to guarantee that the local models reach convergence within each training phase, we set the number of local epochs E to 1 and define the total number of communication rounds R as 200. The hyperparameters for our teacher-student model have already been provided in the **Method** section.

Evaluation Metrics. The evaluation metric is the performance of the model when each domain serves as the test domain. Additionally, we use the average performance **Avg** across all domains to assess the stability of the global model.

Results

We achieved state-of-the-art results on three widely used datasets, particularly on PACS. As observed in Tab.1, both FedDG and DG exhibit strong generalization to nearby domains but struggle with distant domains. For example, domains Photo, Art, and Cartoon are adjacent, so using TTA yields significant benefits, whereas the effect is less pronounced for domain Sketch. Nevertheless, our method, FedSPL, does not lose the ability to recognize domain S during the adaptation process. The sample distribution of the PACS dataset can be referenced from (Zhou et al. 2021).In the OfficeHome Tab.1 and Digit-5 Tab.3 datasets, due to the relatively dispersed data distribution, the effectiveness of TTA methods is not as significant Fig.4 shows the convergence of FedSPL during test-time adaptation. Our FedSPL achieves convergence within just 2 to 3 iterations, demonstrating exceptional stability.

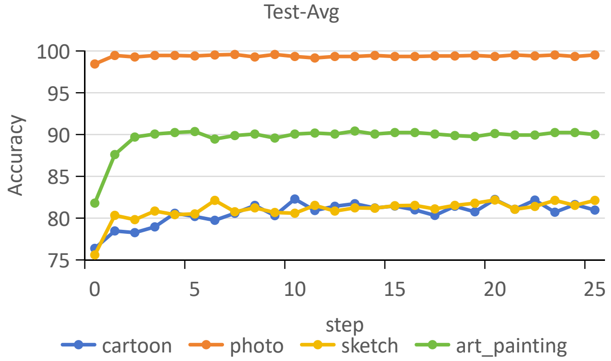

Figure 3: The convergence of the FedSPL after 25 iterations

It is worth noting that, intuitively, one might expect that using TTA methods would always improve model performance. However, our experiments revealed that when the model's initial performance is poor or when the gap between the test unknown domain client and the training domain clients is large, the use of TTA may not yield optimal results and can even lead to a decline in performance.

Ablation Studies

Effectiveness of MBSL and FMR We performed a macro-level ablation study on the two main modules of our model. As observed in Tab.4, both MBSL and FMR contribute to improving the baseline model and complement each other.

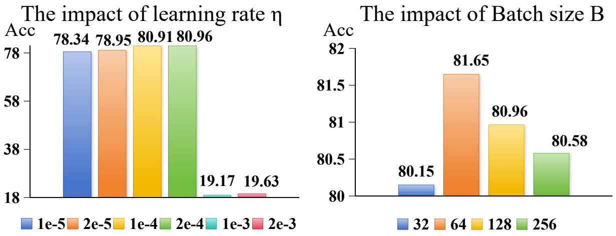

Figure 4: The impact of different learning rates η and batch sizes B on model performance with the PACS Dataset and the Test Set as Domain S.

Additionally, MBSL, as the test adaptation component of FedSPL, also seems to address the global model's inherent learning deficiencies.

$\mathcal{L}_s$	$\mathcal{L}_w$	$\mathcal{L}_c$	P	A	C	S	Avg
×	×	×	92.77	76.67	77.97	81.03	82.26
✓	×	×	97.04	87.55	83.28	**84.27**	88.03
×	✓	×	96.59	84.47	80.38	81.93	85.95
×	×	✓	96.95	85.78	79.31	80.25	85.57
✓	✓	×	97.78	87.74	83.32	83.69	88.13
✓	×	✓	97.37	88.18	83.15	83.18	87.97
×	✓	✓	97.19	85.64	81.74	80.78	86.33
✓	✓	✓	**98.14**	**89.38**	**84.72**	82.84	**88.77**

Table 2: The combination experiments of the three different loss functions were conducted on the PACS dataset using a pre-trained ResNet-18 (Best in **bold**)

Effect of Different Loss Functions In our ablation experiments on different loss functions, we observed that when generalizing to similar domains, methods $\mathcal{L}_s$, $\mathcal{L}_w$, and $\mathcal{L}_c$ all contribute positively to varying degrees as shown in Tab.2. However, when attempting to generalize to domains that are significantly different from the training domains, methods $\mathcal{L}_s$ and $\mathcal{L}_w$ continue to have a positive impact, whereas method $\mathcal{L}_c$ can lead to a decline in model performance. Contrastive learning in method $\mathcal{L}_c$ tends to capture domain-specific features, such as the black-and-white nature of sketch images. Therefore, it may be beneficial to reduce the proportion of $\mathcal{L}_c$ during use. Nevertheless, from an average performance perspective, method $\mathcal{L}_c$ still shows an improvement over the baseline. In our experiments, we also observed that our designed method for selecting strong labels significantly improves model performance. Meanwhile, $\mathcal{L}_w$ and $\mathcal{L}_c$ are specifically designed to prevent extreme cases, thereby enhancing model stability.

Selection of Hyperparameters. FedSPL differs from traditional FedDG in that it continues to learn during the testing process. Therefore, it is crucial to explore the impact of hyperparameters η and B on model performance. As shown in Fig.4, when the learning rate is high, the model fails to converge, but if the learning rate is low, the model performance also suffers. Through our experiments, we found that smaller batch sizes B correspond to better performance, possibly because a smaller B results in a higher learning frequency, leading to improved model performance.We have included some of the experiments in the appendix.

Method	TTA	mt	sv	up	syn	mm	Avg
FedAvg	×	95.47	52.28	89.62	89.75	55.62	76.55
FedProx	×	94.03	59.01	95.20	94.21	61.82	80.85
AM	×	95.77	59.91	95.28	92.10	63.54	81.32
RSC	×	93.12	58.57	93.72	94.76	62.64	80.56
Scaffold	×	91.03	37.15	85.69	71.26	60.33	69.09
GA	×	96.41	64.12	94.01	94.39	62.92	82.37
SHOT	✓	94.69	57.91	89.55	76.43	60.19	75.75
Tent	✓	95.48	60.67	91.67	78.56	62.49	77.77
T3A	✓	94.63	49.90	88.46	75.47	51.25	71.94
MEMO	✓	95.92	52.85	89.84	80.12	55.48	74.84
EM	✓	96.64	57.21	92.29	85.69	62.08	78.78
BBSE	✓	94.47	57.26	91.34	85.54	61.59	78.04
Surgical	✓	97.35	59.93	94.19	86.06	65.87	80.68
ATP	✓	**97.81**	62.18	95.41	87.91	69.98	82.65
FedSPL (ours)	✓	97.70	**78.68**	**96.77**	**94.93**	**70.73**	**87.76**

Table 3: Results (%) of Digit-5 on ResNet18 . (Best in **bold**)

Method	P	A	C	S	Avg
Fedavg	91.73	73.58	76.02	76.46	80.22
FMR	92.77	77.29	77.97	81.03	82.26
MBSL	**99.52**	87.40	82.12	80.96	87.50
FedSPL (ours)	98.14	**89.38**	**84.72**	**82.84**	**88.77**

Table 4: Ablation study with different components of FedSPL. Both FMR and MBSL have demonstrated strong performance, making them indispensable components of FedSPL.

Conclusion

In this paper, we propose a new framework to address FedDG, leveraging test-time adaptation (TTA) methods to enable FedDG models to adapt to the distribution of unknown domains (TTA-FedDG). Based on this framework, we designed the method of Federated domain generalization based on a select Strong Pseudo Label (FedSPL) which combines feature ordering in training and Knowledge Distillation in test time. The FedSPL enhances the model to generalize to unknown clients by incorporating an improved Teacher-Student model into the global model. We introduce a new label selection strategy, and to prevent the selected strong labels from being overly uniform, which could lead to the neglect of other classes, we also design a loss function for weak labels. Additionally, to mitigate the instability issues commonly associated with TTA, we introduce image-based contrastive loss and feature mixing based on feature reordering (FMR) during local model training, thereby improving the robustness of the local models. We have demonstrated the effectiveness of this method across multiple test datasets. Furthermore, our experiments revealed that in FedDG, when the gap between the test unknown domain client and the training domain clients is large, adaptation may struggle to achieve better learning outcomes, and in some cases, performance may even degrade. We hope that our research contributes to enhancing domain generalization in FedDG models while preserving privacy.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant 42401415) and Jiangsu Innovation Capacity Building Program (Project BM2022028).

References

Bao, W.; Wei, T.; Wang, H.; and He, J. 2024. Adaptive test-time personalization for federated learning. *Advances in Neural Information Processing Systems*, 36.

Cao, S.; Liu, Y.; Zheng, J.; Li, W.; Dong, R.; and Fu, H. 2024. Exploring Test-Time Adaptation for Object Detection in Continually Changing Environments. *arXiv preprint arXiv:2406.16439*.

Carlucci, F. M.; D'Innocente, A.; Bucci, S.; Caputo, B.; and Tommasi, T. 2019. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2229–2238.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Du, Y.; Xu, J.; Xiong, H.; Qiu, Q.; Zhen, X.; Snoek, C. G.; and Shao, L. 2020. Learning to learn with variational information bottleneck for domain generalization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, 200–216. Springer.

Hansun, S. 2013. A new approach of moving average method in time series analysis. In *2013 conference on new media studies (CoNMedia)*, 1–4. IEEE.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hershey, J. R.; and Olsen, P. A. 2007. Approximating the Kullback Leibler divergence between Gaussian mixture models. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 4, IV–317. IEEE.

Huang, X.; and Belongie, S. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, 1501–1510.

Huang, Z.; Wang, H.; Xing, E. P.; and Huang, D. 2020. Self-challenging improves cross-domain generalization. In *Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part II 16*, 124–140. Springer.

Jackson, P. T.; Abarghouei, A. A.; Bonner, S.; Breckon, T. P.; and Obara, B. 2019. Style augmentation: data augmentation via style randomization. In *CVPR workshops*, volume 6, 10–11.

Jiang, L.; and Lin, T. 2022. Test-time robust personalization for federated learning. *arXiv preprint arXiv:2205.10920*.

Jiang, M.; Wang, Z.; and Dou, Q. 2022. Harmofl: Harmonizing local and global drifts in federated learning on heterogeneous medical images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 1087–1095.

Jiang, Y.; Wang, Y.; Zhang, R.; Xu, Q.; Zhang, Y.; Chen, X.; and Tian, Q. 2022. Domain-conditioned normalization for test-time domain generalization. In *European Conference on Computer Vision*, 291–307. Springer.

Karim, N.; Mithun, N. C.; Rajvanshi, A.; Chiu, H.-p.; Samarasekera, S.; and Rahnavard, N. 2023. C-sfda: A curriculum learning aided self-training framework for efficient source free domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24120–24131.

Karimireddy, S. P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; and Suresh, A. T. 2020. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, 5132–5143. PMLR.

Kuzborskij, I.; and Orabona, F. 2013. Stability and hypothesis transfer learning. In *International Conference on Machine Learning*, 942–950. PMLR.

Li, D.; Yang, Y.; Song, Y.-Z.; and Hospedales, T. M. 2017. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, 5542–5550.

Li, H.; Li, J.; Guan, X.; Liang, B.; Lai, Y.; and Luo, X. 2019. Research on overfitting of deep learning. In *2019 15th international conference on computational intelligence and security (CIS)*, 78–81. IEEE.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2: 429–450.

Liu, Q.; Chen, C.; Qin, J.; Dou, Q.; and Heng, P.-A. 2021. Feddg: Federated domain generalization on medical image segmentation via episodic learning in continuous frequency space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1013–1023.

Ma, Z.; Zhao, M.; Cai, X.; and Jia, Z. 2021. Fast-convergent federated learning with class-weighted aggregation. *Journal of Systems Architecture*, 117: 102125.

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

Qiao, F.; and Peng, X. 2021. Uncertainty-guided model generalization to unseen domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6790–6800.

Qu, Z.; Li, X.; Duan, R.; Liu, Y.; Tang, B.; and Lu, Z. 2022. Generalized federated learning via sharpness aware minimization. In *International conference on machine learning*, 18250–18280. PMLR.

Shamsian, A.; Navon, A.; Fetaya, E.; and Chechik, G. 2021. Personalized federated learning using hypernetworks. In *International Conference on Machine Learning*, 9489–9502. PMLR.

Sun, Y.; Chong, N.; and Ochiai, H. 2023. Feature distribution matching for federated domain generalization. In *Asian Conference on Machine Learning*, 942–957. PMLR.

Venkateswara, H.; Eusebio, J.; Chakraborty, S.; and Panchanathan, S. 2017. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5018–5027.

Wang, D.; Shelhamer, E.; Liu, S.; Olshausen, B.; and Darrell, T. 2020a. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*.

Wang, F.; and Liu, H. 2021. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2495–2504.

Wang, J.; Liu, Q.; Liang, H.; Joshi, G.; and Poor, H. V. 2020b. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33: 7611–7623.

Wei, Y.; and Han, Y. 2024. Multi-Source Collaborative Gradient Discrepancy Minimization for Federated Domain Generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 15805–15813.

Yu, T.; Bagdasaryan, E.; and Shmatikov, V. 2020. Salvaging federated learning by local adaptation. *arXiv preprint arXiv:2002.04758*.

Zhang, L.; Dong, R.; Yuan, S.; Zhang, J.; Chen, M.; Zheng, J.; and Fu, H. 2024a. DeepLight: Reconstructing High-Resolution Observations of Nighttime Light With Multi-Modal Remote Sensing Data. In Larson, K., ed., *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, 7563–7571. International Joint Conferences on Artificial Intelligence Organization. AI for Good.

Zhang, L.; Ren, Z.; Chen, B.; Gong, P.; Xu, B.; and Fu, H. 2024b. A prolonged artificial nighttime-light dataset of China (1984-2020). *Scientific Data*, 11(1): 414.

Zhang, R.; Xu, Q.; Yao, J.; Zhang, Y.; Tian, Q.; and Wang, Y. 2023. Federated domain generalization with generalization adjustment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3954–3963.

Zhang, Y.; Li, M.; Li, R.; Jia, K.; and Zhang, L. 2022. Exact feature distribution matching for arbitrary style transfer and domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8035–8045.

Zhou, D.; Bousquet, O.; Lal, T.; Weston, J.; and Schölkopf, B. 2003. Learning with local and global consistency. *Advances in neural information processing systems*, 16.

Zhou, K.; Yang, Y.; Qiao, Y.; and Xiang, T. 2021. Domain generalization with mixstyle. *arXiv preprint arXiv:2104.02008*.

Zhou, Q.; Zhang, K.-Y.; Yao, T.; Lu, X.; Ding, S.; and Ma, L. 2024. Test-time domain generalization for face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 175–187.