FRET: Feature Redundancy Elimination for Test Time Adaptation

Supplementary Material

7. Extended Related Work

7.1. Test-Time Adaptation

In realistic scenarios, test data often undergoes natural variations or corruptions, resulting in data distribution shifts between the training and test phases. Recently, various Test-Time Adaptation (TTA) approaches have been proposed to adapt pre-trained models during testing [42, 56]. For batch normalization calibration methods, BN [36] replaces the activation statistics estimated from the training set with those of the test set. For pseudo-labeling methods, TSD [51] filters out unreliable features or predictions with high entropy, as lower entropy typically indicates higher accuracy, and it further filters unreliable samples using a consistency filter. For consistency training methods, TIPI [27] identifies input transformations that can simulate domain shifts and uses regularizers, based on derived bounds, to ensure the network remains invariant to such transformations. For clustering-based training methods, TENT [49] minimizes the prediction entropy of the model on the target data. EATA [28] selects reliable samples to minimize entropy loss during test-time adaptation and uses a Fisher regularizer to stabilize key parameters. The importance of these parameters is estimated from test samples with pseudo labels. SAR [29] removes noisy samples with large gradients and encourages model weights to reach a flat minimum, enhancing robustness against remaining noise. Recently, TEA [57], a state-of-the-art TTA approach, introduces an innovative energy-based perspective to mitigate the effects of distribution changes and has shown advantages over current leading approaches.

7.2. Feature Redundancy Elimination

Feature redundancy is a key concern in both feature extraction and feature selection [26, 43]. Feature extraction methods aim to reduce redundancy by transforming the original feature space into a new low-dimensional feature space while retaining as much relevant information as possible. Two typical feature extraction methods are unsupervised method Principal Component Analysis (PCA) [52] and supervised method Linear Discriminant Analysis (LDA) [9]. The former one performs a linear transformation to create a new feature space where the features are uncorrelated, while the latter one reduces redundancy by identifying feature spaces that best separate different classes by maximizing the between-class dispersion while minimizing within-class dispersion.

Unlike feature extraction, feature selection aims to identify the most representative and non-redundant subset of features from the original feature set [47]. Feature selection methods generally include three strategies: 1) Filter methods [2, 5] use statistical measures (e.g., mutual information, Fisher score) to evaluate feature relevance and redundancy. Features with high relevance are deemed more informative for target variables, while redundant features are removed to enhance feature independence. 2) Wrapper methods [35] assess feature subsets by training models and evaluating their performance. A common wrapper method is Recursive Feature Elimination (RFE) [4], which iteratively removes features and evaluates model performance to identify the optimal subset. 3) Embedded methods [11, 14] integrate feature selection within the model training process, such as Lasso regression [44], which penalizes redundant features during training to optimize model parameters and select the most relevant features. Recently, a novel feature selection approach called SOFT [58] has been proposed, which combines second-order covariance matrices with first-order data matrices by knowledge contrastive distillation for unsupervised feature selection.

7.3. SOFT [58] (Second-Order Unsupervised Feature Selection)

SOFT (Second-Order Unsupervised Feature Selection via Knowledge Contrastive Distillation) [58] proposes a two-stage framework that integrates first-order and second-order information for unsupervised feature selection. Given n samples with d features, it constructs the first-order data matrix $X \in \mathbb{R}^{n\times d}$ and the second-order feature covariance matrix $M_F = X^\top X$. To highlight informative feature interactions, SOFT applies an attention mask θ_M on M_F to produce a refined attention matrix:

$$M_A = M_F \odot \theta_M, \quad M_M = M_F - M_A, \tag{10}$$

where $\odot$ denotes the element-wise product, and M_M is the residual matrix. A symmetric constraint and sparsity regularization are enforced on θ_M via an $\ell_{2,1}$ norm:

$$\mathcal{L}_{2,1} = \|\theta_M\|_{2,1} + \|\theta_M^\top\|_{2,1}. \tag{11}$$

To capture semantic relationships, a shared 2-layer Graph Convolutional Network (GCN) is applied on M_A, M_F, and M_M respectively to obtain attention-based, original, and masked representations. Pseudo labels are generated via PCA and K-means on G_A, guiding a contrastive learning objective that aligns the original representation with attention-guided clustering, and pushes the masked representation away. The final objective integrates all components:

$$\min_{\Theta} \ \mathcal{L}_F + \alpha\mathcal{L}_M + \beta\mathcal{L}_{2,1}, \tag{12}$$

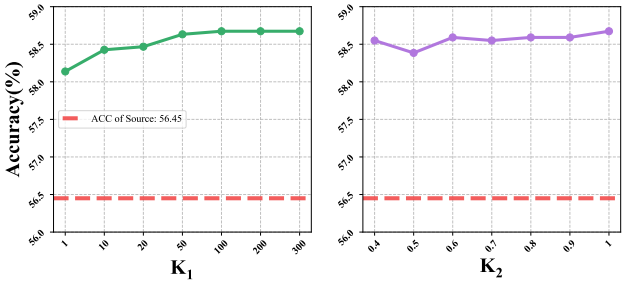

Figure 7. Sensitivity analysis of two G-FRET filtering parameters using ResNet-18 on OfficeHome's Art domain.

where $\mathcal{L}_F$ is a supervised loss with pseudo labels, $\mathcal{L}_M$ is an attention suppression loss for masked features, and $\Theta = \{\theta_M, \theta_G, \theta_C\}$ denotes all trainable parameters.

In the second stage, SOFT constructs a feature graph based on the learned attention matrix M_A, followed by graph segmentation to group correlated features. One representative feature is selected from each cluster, yielding a final subset with reduced redundancy. Unlike traditional ranking-based methods, SOFT emphasizes pairwise relations and joint structural learning, demonstrating strong performance on multiple benchmarks.

8. Some Filter Tricks

Following the previous works [28, 49, 51], to reduce the influence of wrong and noisy results from model g, which may create some incorrect computations, we use Shannon entropy [37] to filter unreliable instances. Specifically, for each class, we only take the representations with the top-K_1 lowest output entropy H_i into computing class centers:

$$H_i = -\sum \sigma(p_i) \log \sigma(p_i) \tag{13}$$

Furthermore, we use the similarity between representation and class centers to generate soft pseudo labels $\hat{y}_i$ for the ith instance, which can be formulated as:

$$\hat{y}_i = \sigma\left([\text{sim}(R_{A_i}, c_1), \text{sim}(R_{A_i}, c_2), \ldots, \text{sim}(R_{A_i}, c_C)]\right) \tag{14}$$

Based on this, we only consider the predictions with the top-K_2 percent lowest output entropy in a data batch while keeping label consistency between pseudo labels and their predictions, i.e., $\text{argmax}(p_i) = \text{argmax}(\hat{y}_i)$, when computing L_R and L_P. We evaluate the effectiveness of these filter tricks on G-FRET. As shown in Fig. 7, G-FRET exhibits insensitivity to K_1, and K_2.

9. Proof of Theoretical Statement

We aim to prove Theorem 1 by extending the generalization bound in Theorem 2 of [55], which considers both the mean and covariance discrepancies between source and target domains.

Theorem 2 in [55]. Let $\mathcal{H}$ be a hypothesis class with VC-dimension d_v. If $\hat{h} \in \mathcal{H}$ minimizes the empirical risk $\hat{\epsilon}_s(h)$ over the source domain $\mathcal{X}_s$, and $h_t^* = \arg\min_{h\in\mathcal{H}} \epsilon_t(h)$ is the optimal hypothesis on the target domain $\mathcal{X}_t$, assuming that all $h \in \mathcal{H}$ are L-Lipschitz continuous, then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$, the following holds:

$$\epsilon_t(\hat{h}) \leq \epsilon_t(h_t^*) + \mathcal{O}\left(\sqrt{\|\mu_s - \mu_t\|_F^2 + \|\Sigma_s - \Sigma_t\|_F^2}\right) + C, \tag{15}$$

where $C = 2\sqrt{\frac{d_v \log(2n_s) - \log(\delta)}{2n_s}} + 2\gamma$, and $\gamma = \min_{h\in\mathcal{H}} \{\epsilon_s(h(t)) + \epsilon_t(h(t))\}$. Here, μ_s, μ_t denote the means of the embeddings in source and target domains, and Σ_s, Σ_t denote the corresponding second-order covariance matrices.

We define the normalized embedding matrix $\tilde{Z} = \frac{Z-\mu}{\sigma}$ such that it has zero mean and unit variance. The empirical covariance matrix of the normalized embeddings is given by:

$$\Sigma = \frac{1}{n} \tilde{Z}^\top \tilde{Z}.$$

To quantify redundancy, we follow the second-order unsupervised feature selection method [58], where the redundancy score R_s is defined as:

$$R_s(Z) = \left\| \tilde{Z}^\top \tilde{Z} - I_d \right\|_1, \tag{16}$$

which measures the deviation of the feature covariance from an identity matrix. A higher R_s indicates stronger linear dependence between features, i.e., higher redundancy.

Note that:

$$\|\Sigma - I_d\|_1 = R_s(Z).$$

Thus, we have the approximation:

$$\begin{aligned} &\|\Sigma_s - \Sigma_t\|_F^2 = \|\Sigma_s - I_d + I_d - \Sigma_t\|_F^2 \\ &\leq \|\Sigma_s - I_d\|_F^2 + \|I_d - \Sigma_t\|_F^2 \approx R_s(Z_s)^2 + R_s(Z_t)^2 \end{aligned}$$

under the assumption that both embeddings are normalized and zero-centered (as done in batchnorm or layernorm), in which case the off-diagonal entries dominate the redundancy.

By substituting the above approximation into Theorem 2 of [55], we obtain:

$$\begin{aligned} &\epsilon_t(\hat{h}) \\ &\leq \epsilon_t(h_t^*) + \mathcal{O}\left(\sqrt{\|\mu_s - \mu_t\|_F^2 + R_s(Z_s)^2 + R_s(Z_t)^2}\right) + C. \end{aligned}$$

Rearranging terms yields the bound in Theorem 1:

$$\epsilon_t(\hat{h}) - \epsilon_t(h_t^*) \leq \mathcal{O}\left(\sqrt{\|\mu_s - \mu_t\|_F^2 + R_s(Z_s)^2 + R_s(Z_t)^2}\right) + C.$$

□

10. Experimental details

10.1. Datasets

We evaluate the performance of the S-FRET and G-FRET on two main tasks: domain generalization and image corruption generalization. Following previous studies, for domain generalization, we use the PACS [21] dataset, consisting of images from seven categories (e.g., objects, animals) across four domains (art paintings, cartoons, photos, and sketches), and the OfficeHome [48] dataset, which includes 65 categories (e.g., office items, home objects) from four domains (art, clipart, product, and real-world images). For image corruption generalization, we utilize the CIFAR10-C , CIFAR100-C and ImageNet-C [13] datasets, which contain 15 types of corruption at five severity levels. To be consistent with prior research [20], all experiments are conducted at the highest severity level (level 5). To implement label shifts, we adjust the CIFAR-100-C datasets to follow a long-tail distribution [7], denoted as CIFAR-100-C-LT.

10.2. Comparison Methods

The comparison methods we employ include the non-adaptive source model and four baseline methods which are commonly used in several studies: BN [36], TENT [49], EATA [28], and SAR [29]. Additionally, we adopt three recently proposed state-of-the-art methods: TSD [51], TIPI [27], and TEA [57].

10.3. Models and Implementation

For domain generalization tasks, we use ResNet-18/50 [12] with batch normalization as the backbone network. These networks are pretrained on data from three domains and then tested on the remaining domain.

For image corruption, we use ResNet-18 as the backbone network. We train it on the clean versions of the CIFAR-10 and CIFAR-100 datasets, and for ImageNet-C, we leverage pre-trained model from TorchVision. When it comes to CIFAR-10/100-C, we apply all 15 corruption types sequentially to assess continuous adaptation capabilities. For ImageNet-C, we apply each of the 15 corruption types independently to the adapted model.

To ensure fairness, we report mean and standard deviation of 5 runs with different random seeds (0, 1, 2, 3, 4). In addition, we set the batch size as 128 during testing and independently perform hyperparameter tuning for each method to achieve the highest possible accuracy as the final result. All implementations are carried out using PyTorch [33] and executed on a single NVIDIA 4090 GPU for all experiments.

10.4. Complex/Large Dataset

VLCS [45] contains four domains: Caltech101, LabelMe, SUN09, and VOC2007, with a total of 10,729 images across 5 classes. The label distribution across the domains in VLCS exhibits substantial variation , which might be a contributing factor to the poor performance of most Test-Time Adaptation methods on this dataset [51].

DomainNet [34] is a large-scale dataset used in transfer learning, consisting of six domains: Clipart, Infograph, Painting, Quickdraw, Real, and Sketch. It consists of a total of 586,575 images, with each domain containing 345 classes.

ImageNet-C [13] is significantly larger compared to the CIFAR10-C and CIFAR100-C. CIFAR10-C and CIFAR100-C consist of 50,000 training images and 10,000 test images each, divided into 10 and 100 classes respectively. In contrast, ImageNet-C contains 1,281,167 training images and 50,000 test images, categorized into 1,000 classes. Specifically, ImageNet-C encompasses 15 types of corruption with five levels of severity. In our experiments, we employed the highest corruption level (level 5). For the pre-trained model on ImageNet-C, we utilize the model provided by TorchVision.

10.5. Experiment Setting Details

For hyper-parameter selection in Domain Generalization task (Sec. 5.2.1), we first identify the optimal parameter set based on the highest accuracy achieved on the default domain (art paintings in PACS and art in OfficeHome). These parameters are then applied to other domains to assess their performance. Specifically, we conduct a search for the learning rate within the range {1e-7, 5e-7, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4, 5e-4, 1e-3, 5e-3, 1e-2, 5e-2}. For methods that include an entropy filter component (e.g., TSD, G-FRET), we explore the entropy filter hyperparameter in the set {1, 5, 10, 15, 20, 50, 100, 200, 300}. For the G-FRET, we perform hyperparameter tuning for λ within the range {1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 1e-1, 1, 10, 100} and K_2 within the values {0.5, 0.6, 0.7, 0.8, 0.9, 1}.

For the Image Corruption task (Sec. 5.2.2), each Test-Time Adaptation (TTA) method continuously adapts to 15 types of image corruptions in the specified order for CIFAR-10/100-C: [Gaussian Noise, Shot Noise, Impulse Noise, Defocus Blur, Glass Blur, Motion Blur, Zoom Blur, Snow, Frost, Fog, Brightness, Contrast, Elastic Transformation, Pixelate, JPEG Compression]. However, for ImageNet-C, we adopt a strategy of independently adapting to each of the 15 corruption types separately. The hyperparameter ranges

remain consistent with those utilized in Domain Generalization. The best performance results obtained for each method are selected as the final experimental outcomes.

11. Additional Experimental Results

11.1. Detailed Results Across Five Random Seeds

To ensure the fairness of our evaluation, we conduct experiments using five different random seeds (0, 1, 2, 3, and 4). The detailed results corresponding to each random seed are presented in Tabs. 7 to 11, which highlights the robustness and consistency of our proposed methods S-FRET and G-FRET.

11.2. Detailed Results for Image Corruption

In this section, we provide a complete listing of comparisons between S-FRET, G-FRET, and other state-of-the-art methods for Image Corruption (Sec. 5.2.2) on CIFAR-10/100-C and ImageNet-C datasets at damage level of 5, as shown in Tabs. 12 to 14.

11.3. Detailed Results for tSNE Experiment

In this section, we supplement the t-SNE visualizations of embedded features (Sec. 5.3.3) across 15 corruption types in the CIFAR-10-C dataset using ResNet-18 and ViT-B/16 models, with the specific results presented in Figs. 8 and 9.

11.4. Detailed Results for Scalability Experiment

In the Scalability experiment (Sec. 5.4.3), we validate our methods using larger and more complex datasets including VLCS, DomainNet, and ImageNet-C, as well as on the ViT backbone, to demonstrate that our approach can robustly improve performance across diverse datasets and different backbones.

For VLCS and DomainNet, we employ hyperparameter selection within the same range as the Domain Generalization task. However, unlike the Domain Generalization task, we independently selected hyperparameters for each domain rather than applying the parameters from the default domain to others.

For ImageNet-C, we adapt the TTA method to each corruption type individually. We select hyperparameters optimized for the default corruption type (Gaussian Noise), and applied these parameters to other corruption types. The detailed results are presented in Tab. 15, Tab. 16 , and Tab. 17 .

Figure 8. Discriminability visualization of embedded features before and after adaptation via S-FRET and G-FRET, on 15 corruption types of the CIFAR-10-C dataset using ResNet-18.

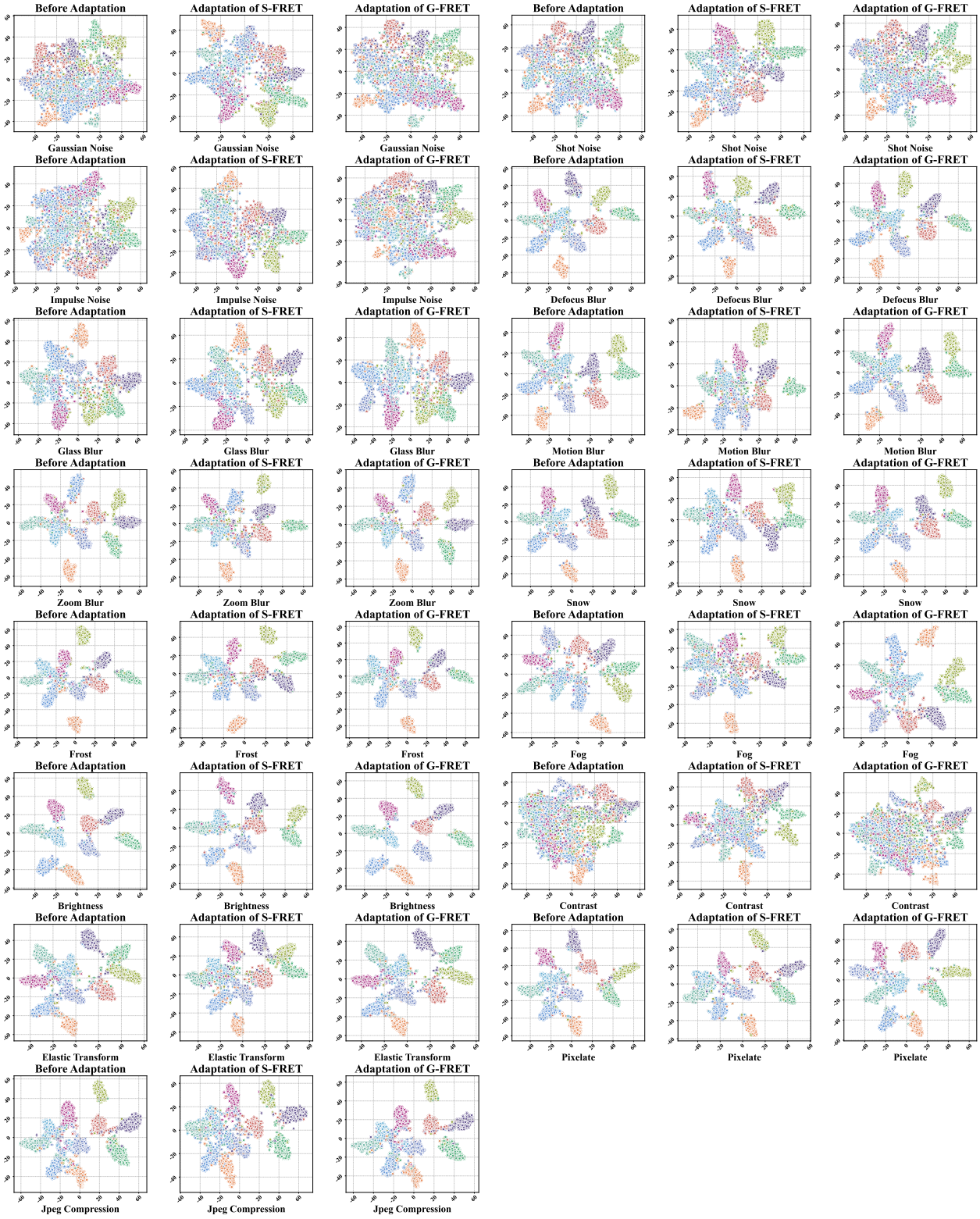

Figure 9. Discriminability visualization of embedded features before and after adaptation via S-FRET and G-FRET, on 15 corruption types of the CIFAR-10-C dataset using ViT-B/16.

Backbone	Method	PACS				Avg	OfficeHome				Avg
		A	C	P	S		A	C	P	R	
ResNet-18	Source [12]	78.37	77.39	95.03	76.58	81.84	56.45	48.02	71.34	72.23	62.01
	BN [36]	80.91	80.80	95.09	73.84	82.66	55.62	49.32	70.60	72.66	62.05
	SAR [29]	83.30	82.17	95.09	79.69	85.06	57.15	50.31	70.24	72.34	62.51
	EATA [28]	82.71	81.53	94.91	74.19	83.34	56.41	49.62	71.66	72.27	62.49
	TENT [49]	82.76	82.68	95.33	78.19	84.74	56.94	50.65	71.86	72.92	63.09
	TSD [51]	**86.96**	86.73	96.41	81.22	87.83	58.06	49.81	71.37	70.67	62.47
	TEA [57]	86.47	85.79	95.69	80.81	87.19	**58.63**	50.56	71.95	72.92	63.52
	TIPI [27]	85.50	84.90	96.05	**83.13**	87.39	57.03	50.61	72.07	**73.28**	63.25
	S-FRET	86.28	86.69	96.35	74.22	85.88	56.20	50.08	71.57	72.64	62.62
	G-FRET	86.82	**87.03**	**96.65**	81.29	**87.95**	57.73	**51.36**	**73.10**	72.99	**63.79**
ResNet-50	Source [12]	83.89	81.02	96.17	78.04	84.78	64.85	52.26	75.04	75.88	67.01
	BN [36]	85.50	85.62	96.77	72.05	84.99	63.54	52.71	73.89	75.05	66.30
	SAR [29]	85.55	85.62	96.77	75.24	85.79	64.77	55.92	75.24	75.81	67.94
	EATA [28]	84.67	85.20	96.35	72.36	84.64	63.91	54.04	74.72	75.51	67.05
	TENT [49]	88.09	87.33	97.19	79.69	88.07	64.61	54.80	75.06	76.20	67.67
	TSD [51]	90.43	89.89	**97.84**	81.80	89.99	65.27	56.77	76.19	76.41	68.66
	TEA [57]	88.09	87.88	97.49	81.39	88.71	66.25	**57.50**	75.20	76.68	68.91
	TIPI [27]	88.18	87.93	97.13	78.80	88.01	64.73	56.24	75.47	77.00	68.36
	S-FRET	89.99	89.51	**97.84**	76.30	88.41	64.15	54.50	75.74	76.25	67.66
	G-FRET	**90.72**	**90.15**	**97.84**	**82.29**	**90.25**	**66.42**	57.11	**76.21**	**77.35**	**69.27**

Table 7. At random seed 0, the accuracy comparison of different TTA methods on PACS and OfficeHome datasets based on ResNet-18 and ResNet-50 backbones. The best results are highlighted in **boldface**, and the second ones are underlined.

Backbone	Method	PACS				Avg	OfficeHome				Avg
		A	C	P	S		A	C	P	R	
ResNet-18	Source [12]	78.37	77.39	95.03	76.58	81.84	56.45	48.02	71.34	72.23	62.01
	BN [36]	81.10	80.59	95.33	73.84	82.71	55.83	48.80	70.78	72.21	61.90
	SAR [29]	84.33	80.03	95.33	79.59	84.82	56.90	50.22	70.06	72.89	62.52
	EATA [28]	82.96	82.47	95.45	73.00	83.47	57.03	50.13	71.03	72.96	62.79
	TENT [49]	84.91	81.19	95.57	82.82	86.12	57.77	50.22	72.56	72.62	63.29
	TSD [51]	86.62	86.39	95.87	81.50	87.59	57.52	49.26	72.18	71.20	62.54
	TEA [57]	**87.79**	86.60	95.99	82.21	88.15	56.78	**50.70**	72.02	72.85	63.09
	TIPI [27]	85.11	83.23	95.87	**85.03**	87.31	**57.81**	50.19	72.56	72.96	63.38
	S-FRET	85.84	86.69	95.93	72.84	85.33	56.53	49.51	71.71	72.53	62.57
	G-FRET	87.26	**86.99**	**96.59**	82.74	**88.39**	57.23	50.61	**73.51**	**73.17**	**63.63**
ResNet-50	Source [12]	83.89	81.02	96.17	78.04	84.78	64.85	52.26	75.04	75.88	67.01
	BN [36]	85.01	85.88	96.65	71.88	84.85	63.00	53.54	73.60	74.96	66.27
	SAR [29]	85.01	85.88	96.65	75.92	85.86	65.06	56.31	74.50	76.45	68.08
	EATA [28]	85.35	85.41	96.71	72.23	84.92	64.81	53.88	73.80	75.60	67.02
	TENT [49]	86.23	86.95	97.01	79.77	87.49	64.81	55.58	74.75	76.38	67.88
	TSD [51]	**89.75**	89.63	**97.49**	83.76	90.16	65.72	57.14	76.39	76.18	68.86
	TEA [57]	89.16	88.05	96.89	82.54	89.16	65.22	57.69	75.67	**77.37**	68.99
	TIPI [27]	87.16	86.99	97.07	80.99	88.05	65.39	56.11	75.69	76.27	68.36
	S-FRET	89.36	89.46	97.37	78.57	88.69	64.61	55.28	75.60	76.50	68.00
	G-FRET	**89.75**	**89.97**	**97.49**	**85.52**	**90.68**	**66.17**	**57.87**	**76.93**	76.80	**69.44**

Table 8. At random seed 1, the accuracy comparison of different TTA methods on PACS and OfficeHome datasets based on ResNet-18 and ResNet-50 backbones. The best results are highlighted in **boldface**, and the second ones are underlined.

Backbone	Method	PACS				Avg	OfficeHome				Avg
		A	C	P	S		A	C	P	R	
ResNet-18	Source [12]	78.37	77.39	95.03	76.58	81.84	56.45	48.02	71.34	72.23	62.01
	BN [36]	81.05	80.42	94.97	73.73	82.54	55.71	49.10	70.85	72.55	62.05
	SAR [29]	83.69	82.08	94.97	81.01	85.44	57.11	50.70	71.03	72.76	62.90
	EATA [28]	80.91	81.06	94.97	72.77	82.43	56.49	49.10	71.95	73.22	62.69
	TENT [49]	81.20	83.70	95.51	81.42	85.46	57.19	50.42	71.80	73.08	63.12
	TSD [51]	87.06	87.12	95.93	83.07	88.29	**57.48**	51.11	71.23	70.32	62.54
	TEA [57]	87.09	**87.97**	**96.47**	81.50	88.26	56.53	50.42	71.91	72.96	62.96
	TIPI [27]	82.86	84.04	95.93	**84.58**	86.85	57.19	50.49	72.00	73.45	63.28
	S-FRET	86.28	86.35	95.69	78.34	86.66	56.04	49.46	71.64	72.89	62.51
	G-FRET	**87.45**	87.07	96.35	83.66	**88.63**	57.23	**51.59**	**72.85**	**73.84**	**63.88**
ResNet-50	Source [12]	83.89	81.02	96.17	78.04	84.78	64.85	52.26	75.04	75.88	67.01
	BN [36]	84.72	85.20	96.59	72.72	84.80	63.33	53.08	73.64	74.82	66.22
	SAR [29]	84.72	85.20	96.59	75.18	85.42	64.73	56.93	74.86	76.15	68.17
	EATA [28]	85.45	85.11	**96.47**	72.38	84.85	63.86	54.73	74.00	75.60	67.05
	TENT [49]	87.06	85.92	96.65	75.34	86.24	64.32	55.60	74.09	76.02	67.51
	TSD [51]	**91.85**	**89.76**	97.49	79.92	**89.75**	64.89	57.46	75.76	76.54	68.66
	TEA [57]	88.57	88.27	97.07	**80.48**	88.60	65.93	57.00	75.96	76.25	68.78
	TIPI [27]	**87.45**	85.41	96.89	77.53	86.82	64.85	56.98	75.02	76.41	68.31
	S-FRET	90.77	88.69	97.31	77.07	88.46	64.44	55.40	75.26	76.29	67.85
	G-FRET	**91.85**	89.59	**97.54**	78.42	89.35	**66.05**	**57.73**	**76.17**	**77.25**	**69.30**

Table 9. At random seed 2, the accuracy comparison of different TTA methods on PACS and OfficeHome datasets based on ResNet-18 and ResNet-50 backbones. The best results are highlighted in **boldface**, and the second ones are underlined.

Backbone	Method	PACS				Avg	OfficeHome				Avg
		A	C	P	S		A	C	P	R	
ResNet-18	Source [12]	78.37	77.39	95.03	76.58	81.84	56.45	48.02	71.34	72.23	62.01
	BN [36]	80.96	80.46	95.03	74.12	82.64	55.75	49.67	70.76	72.64	62.20
	SAR [29]	83.40	81.02	95.03	81.01	85.11	57.27	50.68	70.80	73.08	62.96
	EATA [28]	82.23	80.84	94.97	72.31	82.59	56.32	49.97	71.53	72.57	62.60
	TENT [49]	83.15	79.91	95.27	82.08	85.10	57.03	50.77	72.09	73.40	63.32
	TSD [51]	86.77	86.52	95.93	83.61	88.21	**57.85**	49.85	71.37	71.52	62.65
	TEA [57]	**87.26**	86.26	**96.59**	83.61	88.43	56.70	50.42	71.75	72.69	62.89
	TIPI [27]	84.57	83.40	95.57	83.69	86.81	56.94	50.72	72.11	73.26	63.26
	S-FRET	86.18	86.35	96.05	75.82	86.10	56.12	50.36	71.68	72.80	62.74
	G-FRET	86.87	**87.33**	95.87	**84.07**	**88.53**	57.31	**51.64**	**72.76**	**73.65**	**63.84**
ResNet-50	Source [12]	83.89	81.02	96.17	78.04	84.78	64.85	52.26	75.04	75.88	67.01
	BN [36]	85.16	85.20	96.71	72.18	84.81	63.41	52.88	73.26	75.58	66.28
	SAR [29]	85.40	85.20	96.71	76.86	86.04	64.73	56.77	74.75	76.50	68.19
	EATA [28]	85.21	85.49	96.41	72.10	84.80	64.15	53.36	74.41	75.42	66.83
	TENT [49]	86.18	86.90	96.95	79.97	87.50	64.32	55.51	74.93	76.27	67.76
	TSD [51]	90.48	**90.32**	97.78	81.98	90.14	65.31	56.59	75.74	76.75	68.60
	TEA [57]	88.13	86.95	97.07	82.41	88.64	65.27	**58.56**	76.23	**77.39**	**69.36**
	TIPI [27]	86.77	87.54	97.07	81.04	88.10	64.65	56.72	75.20	77.09	68.42
	S-FRET	89.31	88.95	97.72	78.16	88.54	64.61	55.10	75.38	76.59	67.92
	G-FRET	**90.82**	90.23	**97.90**	**83.46**	**90.60**	**65.47**	57.34	**76.28**	77.32	69.10

Table 10. At random seed 3, the accuracy comparison of different TTA methods on PACS and OfficeHome datasets based on ResNet-18 and ResNet-50 backbones. The best results are highlighted in **boldface**, and the second ones are underlined.

Backbone	Method	PACS				Avg	OfficeHome				Avg
		A	C	P	S		A	C	P	R	
ResNet-18	Source [12]	78.37	77.39	95.03	76.58	81.84	56.45	48.02	71.34	72.23	62.01
	BN [36]	81.10	81.06	95.27	73.63	82.77	55.38	49.28	70.65	72.39	61.92
	SAR [29]	83.40	82.04	95.27	80.91	85.40	58.06	50.79	71.39	72.66	63.23
	EATA [28]	81.49	79.65	95.51	74.68	82.83	55.87	49.92	71.34	73.22	62.59
	TENT [49]	85.69	83.83	96.05	80.73	86.58	57.31	50.17	72.61	**73.42**	63.38
	TSD [51]	**87.84**	**87.93**	95.75	83.41	88.73	**58.10**	50.22	71.10	70.87	62.57
	TEA [57]	86.57	86.05	**96.13**	82.72	87.87	56.70	50.22	71.57	72.85	62.83
	TIPI [27]	85.89	85.62	95.87	83.84	87.80	57.19	50.10	72.40	**73.42**	63.28
	S-FRET	86.67	87.24	95.81	78.29	87.00	56.41	49.90	71.66	72.71	62.67
	G-FRET	87.60	87.54	95.93	**85.06**	**89.03**	57.97	**51.20**	**72.99**	73.38	**63.89**
ResNet-50	Source [12]	83.89	81.02	96.17	78.04	84.78	64.85	52.26	75.04	75.88	67.01
	BN [36]	85.55	85.58	96.29	72.41	84.96	63.21	52.90	73.55	75.07	66.18
	SAR [29]	85.55	85.58	96.29	76.89	86.08	64.61	55.56	74.79	75.97	67.73
	EATA [28]	85.01	85.58	96.77	72.26	84.90	64.52	54.46	74.30	76.47	67.44
	TENT [49]	87.79	86.69	96.83	79.41	87.68	64.61	54.27	74.59	76.11	67.39
	TSD [51]	91.65	**90.10**	97.19	80.33	89.82	65.47	57.00	76.53	76.73	68.93
	TEA [57]	88.57	87.59	**97.37**	80.40	88.48	65.43	57.02	75.92	76.54	68.73
	TIPI [27]	88.53	86.99	96.89	79.94	88.09	64.98	56.29	75.20	76.91	68.34
	S-FRET	90.09	88.40	97.07	77.14	88.17	64.28	55.03	75.83	76.45	67.90
	G-FRET	**91.80**	90.06	97.19	**81.93**	**90.24**	**66.50**	**58.08**	**76.71**	**77.23**	**69.63**

Table 11. At random seed 4, the accuracy comparison of different TTA methods on PACS and OfficeHome datasets based on ResNet-18 and ResNet-50 backbones. The best results are highlighted in **boldface**, and the second ones are underlined.

Method	t ⟶															Avg
	Gau.	Sho.	Imp.	Def.	Gla.	Mot.	Zoo.	Sno.	Fro.	Fog	Bri.	Con.	Ela.	Pix.	Jpe.	
Source [12]	27.43	33.56	21.57	43.64	40.48	51.26	51.29	68.18	54.52	66.65	87.50	27.59	67.06	48.86	72.37	50.80
BN [36]	66.30	68.18	57.13	82.50	57.44	79.73	81.98	74.83	74.12	78.91	86.96	82.02	70.23	73.43	70.94	73.65
TENT [49]	67.26	71.46	61.21	84.07	61.37	81.66	84.36	78.18	77.55	80.14	88.44	81.41	73.54	78.53	76.19	76.36
EATA [28]	66.39	68.50	57.32	82.52	57.42	79.94	82.09	74.80	74.14	78.90	86.98	81.93	70.10	73.62	70.88	73.70
SAR [29]	66.46	68.24	57.47	82.52	57.83	79.76	81.98	74.83	74.29	78.92	86.96	82.36	70.26	73.43	70.94	73.75
TIPI [27]	67.57	**72.14**	**62.88**	84.19	**63.55**	81.63	84.44	**79.06**	**79.07**	79.61	88.68	81.92	**75.33**	**79.92**	**78.11**	77.21
TEA [57]	66.71	69.24	59.46	82.78	59.98	80.87	82.88	76.40	75.60	79.82	86.72	81.48	71.89	74.91	72.91	74.78
TSD [51]	66.97	70.31	60.63	83.24	61.10	81.52	83.97	77.15	76.75	80.08	86.76	80.42	72.66	76.43	73.42	75.43
S-FRET	66.52	69.45	59.40	82.78	59.43	80.64	82.98	76.05	75.73	79.59	86.92	80.68	71.56	75.11	72.61	74.63
G-FRET	**67.79**	71.83	62.81	**84.31**	62.63	**82.07**	**84.89**	78.91	79.00	**81.01**	**88.87**	**82.69**	74.86	79.47	77.74	**77.26**

Table 12. Accuracy comparisons of different TTA methods on CIFAR-10-C dataset at damage level of 5, with 15 types of damage applied sequentially to a continuously adapted model. The best results are highlighted in **boldface**, and the second ones are underlined.

Method	t →															Avg
	Gau.	Sho.	Imp.	Def.	Gla.	Mot.	Zoo.	Sno.	Fro.	Fog	Bri.	Con.	Ela.	Pix.	Jpe.	
Source [12]	10.46	12.49	3.36	34.44	23.63	38.10	42.67	39.25	33.01	32.84	55.78	11.55	46.48	34.88	46.15	31.01
BN [36]	39.84	39.51	29.88	56.43	41.08	54.34	58.82	48.52	49.36	46.60	61.82	48.78	49.92	54.17	45.39	48.30
TENT [49]	40.61	41.94	32.09	57.84	44.35	56.57	60.96	51.51	52.03	49.23	**63.94**	49.62	53.53	**57.60**	50.03	50.79
EATA [28]	40.59	41.82	32.58	**57.97**	43.43	56.76	60.33	50.79	51.90	48.71	63.83	49.88	53.96	**57.60**	50.50	50.71
SAR [29]	40.09	40.67	31.52	57.01	42.16	55.63	59.54	50.53	50.31	47.73	62.50	43.27	51.03	55.49	48.80	49.09
TIPI [27]	40.62	42.29	32.67	57.06	**44.84**	55.45	59.58	**52.19**	52.15	46.33	61.91	43.60	52.20	57.39	50.67	49.93
TEA [57]	40.13	39.90	30.82	56.28	41.48	54.73	59.16	48.79	49.31	46.26	61.41	48.48	50.22	54.03	46.63	48.51
TSD [51]	39.84	39.65	30.14	56.63	41.17	54.65	59.03	48.71	49.71	47.25	61.95	48.84	50.65	54.78	46.36	48.62
S-FRET	39.84	39.94	31.10	56.66	42.01	55.19	59.46	48.66	49.21	47.21	61.52	46.80	50.66	53.88	45.75	48.53
G-FRET	**41.08**	**43.23**	**33.69**	57.49	44.07	**56.87**	**61.16**	51.24	**52.18**	**49.25**	63.12	**50.04**	**54.09**	57.42	**51.02**	**51.06**

Table 13. Accuracy comparisons of different TTA methods on CIFAR-100-C dataset at damage level of 5, with 15 types of damage applied sequentially to a continuously adapted model. The best results are highlighted in **boldface**, and the second ones are underlined.

Method	ImageNet-C															Avg
	Gau.	Sho.	Imp.	Def.	Gla.	Mot.	Zoo.	Sno.	Fro.	Fog	Bri.	Con.	Ela.	Pix.	Jpe.	
Source [12]	1.54	2.27	1.48	11.44	8.68	11.12	17.62	10.64	16.21	14.02	51.52	3.44	16.49	23.35	30.67	14.70
BN [36]	13.65	14.84	14.17	11.95	13.04	23.34	33.89	29.18	28.42	40.80	58.11	12.09	38.92	44.35	37.08	27.59
TENT [49]	23.45	25.71	24.08	18.79	20.90	33.54	42.85	39.64	32.95	50.36	60.13	10.68	48.81	51.96	46.98	35.39
EATA [28]	**28.24**	**30.16**	**28.88**	**25.30**	**25.74**	**36.61**	43.71	**41.80**	36.42	50.87	59.12	**31.75**	49.10	52.33	47.82	**39.19**
SAR [29]	28.04	29.59	27.88	23.66	23.90	36.16	43.40	40.94	**36.71**	**51.01**	60.18	27.38	48.95	**52.47**	**47.98**	38.55
TIPI [27]	24.45	26.52	24.75	20.37	22.25	33.65	42.46	39.31	33.47	49.93	59.44	12.53	48.41	51.51	46.92	35.73
TEA [57]	18.82	20.50	19.00	16.27	17.68	28.51	39.17	35.19	32.26	46.92	59.16	15.42	44.39	48.81	43.64	32.38
TSD [51]	15.60	16.99	16.13	15.59	15.41	28.69	38.07	32.92	30.01	45.90	58.69	7.62	41.06	47.47	41.52	30.11
S-FRET	15.04	16.40	15.57	13.77	14.67	25.65	36.00	31.08	29.08	43.13	58.64	12.33	40.37	45.76	39.07	29.10
G-FRET	24.85	27.47	25.49	20.82	22.71	35.10	**43.76**	40.66	36.68	50.80	**60.3**	14.20	**49.28**	52.24	47.52	36.79

Table 14. Accuracy comparisons of different TTA methods on ImageNet-C dataset at damage level of 5, with 15 types of damage applied independently to the adapted model based on ResNet-18. The best results are highlighted in **boldface**, and the second ones are underlined.

Method	VLCS				Avg
	C	L	S	V	
ResNet-18	94.49	60.96	67.73	71.50	73.67
+S-FRET	96.11	59.71	67.28	72.51	73.90
+G-FRET	**96.68**	**64.42**	**67.82**	**73.99**	**75.73**
ResNet-50	95.55	**60.77**	71.12	72.16	74.90
+S-FRET	96.05	57.89	69.86	**78.10**	75.48
+G-FRET	**96.96**	58.51	**72.27**	77.93	**76.42**
ViT-B/16	97.81	64.38	69.71	73.84	76.44
+S-FRET	97.81	67.32	69.59	73.99	77.18
+G-FRET	**98.52**	**68.00**	**73.49**	**74.23**	**78.56**

Table 15. Accuracy on the VLCS dataset with different backbones: ResNet-18/50 and ViT-B/16.

Method	DomainNet						Avg
	C	I	P	Q	R	S	
ResNet-18	57.30	**16.86**	45.03	12.69	56.89	46.00	39.13
+S-FRET	57.69	12.58	44.55	15.18	**57.71**	47.86	39.26
+G-FRET	**58.97**	14.10	**46.16**	**15.22**	57.42	**49.48**	**40.22**
ResNet-50	63.68	**20.93**	50.35	12.95	62.16	51.42	43.58
+S-FRET	63.95	15.72	50.00	**15.23**	**63.51**	53.09	43.59
+G-FRET	**64.85**	17.61	**51.19**	14.71	63.33	**53.61**	**44.22**
ViT-B/16	71.91	25.56	55.95	18.36	70.66	57.45	49.98
+S-FRET	72.30	**27.17**	**59.45**	17.25	71.48	59.65	51.22
+G-FRET	**72.63**	26.04	58.28	**18.92**	**72.61**	**60.50**	**51.50**

Table 16. Accuracy on the DomainNet dataset with different backbones: ResNet-18/50 and ViT-B/16.

Method	ImageNet-C															Avg
	Gau.	Sho.	Imp.	Def.	Gla.	Mot.	Zoo.	Sno.	Fro.	Fog	Bri.	Con.	Ela.	Pix.	Jpe.	
ResNet-18	1.54	2.27	1.48	11.44	8.68	11.12	17.62	10.64	16.21	14.02	51.52	3.44	16.49	23.35	30.67	14.70
+S-FRET	15.04	16.40	15.57	13.77	14.67	25.65	36.00	31.08	29.08	43.13	58.64	12.33	40.37	45.76	39.07	29.10
+G-FRET	**24.85**	**27.47**	**25.49**	**20.82**	**22.71**	**35.10**	**43.76**	**40.66**	**36.68**	**50.80**	**60.30**	**14.20**	**49.28**	**52.24**	**47.52**	**36.79**
ResNet-50	3.00	3.70	2.64	17.91	9.74	14.71	22.45	16.60	23.06	24.01	59.12	5.38	16.51	20.87	32.63	18.15
+S-FRET	19.26	18.81	19.91	18.66	18.16	29.96	43.32	38.29	34.02	51.74	66.52	16.69	47.22	52.51	44.60	34.65
+G-FRET	**29.22**	**29.13**	**29.83**	**25.99**	**26.68**	**44.10**	**50.83**	**49.15**	**43.40**	**58.43**	**67.35**	**17.88**	**57.12**	**59.68**	**53.76**	**42.84**
ViT-B/16	35.09	32.16	35.88	31.42	25.31	39.45	31.55	24.47	30.13	54.74	64.48	48.98	34.20	53.17	56.45	39.83
+S-FRET	51.62	53.02	53.54	49.54	50.21	56.68	**59.19**	**61.93**	**61.22**	70.65	72.85	65.43	66.11	67.87	65.61	60.36
+G-FRET	**57.56**	**56.80**	**58.02**	**56.86**	**57.42**	**62.11**	58.98	41.25	59.39	**72.69**	**77.31**	**70.33**	**66.61**	**71.93**	**70.25**	**62.50**

Table 17. Accuracy on the ImageNet-C dataset with different backbones: ResNet-18/50 and ViT-B/16.