



Full Length Article

Stable multi-view representation learning for out-of-distribution classification

Zheng Zhang^{a,1}, Yongxi Yang^{a,1}, Yan Chen^b, Bingbing Jiang^c, Liang Du^d, Peng Zhou^{a,*}^a Anhui Provincial International Joint Research Center for Advanced Technology in Medical Imaging, School of Computer Science and Technology, Anhui University, Hefei, 230601, China^b College of Computer Science and Technology, Taiyuan University of Technology, Taiyuan, 030600, China^c School of Information Science and Technology, Hangzhou Normal University, Hangzhou, 400065, China^d School of Computer and Information Technology, Shanxi University, Taiyuan, 030006, China

ARTICLE INFO

Keywords:

Multi-view fusion

Stable learning

Out-of-distribution generalization

ABSTRACT

Multi-view classification is an important task in information fusion and attracts increasing attention. Although conventional multi-view classification methods have achieved promising performance, they often have an independent and identically distributed assumption on the training and test set, and thus they can only work under the in-distribution condition. However, in real-world applications, especially handling the low-quality data, the in-distribution condition is often unsatisfied, leading to the out-of-distribution (OOD) problem. To tackle the OOD problem, we propose a novel stable multi-view representation learning framework in this paper. In more detail, in the multi-view representation learning, we carefully design an adaptive dual-weighting mechanism that dynamically learns the weights of samples and dimensions simultaneously. Due to this adaptive dual-weighting mechanism, we can eliminate spurious correlations in the representation and enforce the multi-view classifier to focus on the true correlations between causal features and labels rather than spurious correlations. Extensive experiments on benchmark OOD datasets by comparing with state-of-the-art multi-view classification methods and OOD classification methods demonstrate the effectiveness and superiority of our proposed method. The source code is publicly available at: <https://github.com/zz978233034/stable-multi-view-classification>.

1. Introduction

In many information fusion applications, data are often represented by multiple views, which are so-called multi-view data [1,2]. Multi-view learning tries to learn the consensus information from the multiple views and is widely studied in the information fusion community [3,4]. For example, Lou et al. [5] proposed a structure-enhanced self-supervised weighted information bottleneck method for multi-view clustering, which adaptively learns the importance of different views by incorporating both view-contained information and pseudo-label-based self-supervision. Wan et al. [6] proposed a dynamic multi-view labeling strategy with shared and specific information.

However, real-world multi-view data, especially the low-quality multi-view data, are rarely collected under ideal conditions. Different views may suffer from heterogeneous forms of low-quality information, such as noisy and unreliable views [7,8]. These issues not only weaken the complementarity among views, but also introduce

spurious correlations between view-specific features and labels [9]. As a result, the learned representations may become overly dependent on unstable information in the training data, leading to unreliable fusion and poor generalization when the data distribution changes.

Under such low-quality and distributionally uncertain scenarios, multi-view classification becomes particularly challenging. Existing multi-view classification methods are often based on the independent and identically distributed (i.i.d.) assumption, i.e., the training data and test data are sampled independently from the same distribution [10]. Although these methods have achieved promising performance, this assumption may not be satisfied in real-world applications, where the data distribution may shift due to view heterogeneity, measurement noise, missing or inconsistent views, and other uncontrolled factors [11–13]. Traditional multi-view classification methods may therefore fail to provide reliable predictions on low-quality data with uncontrolled or even unknown distributions.

* Corresponding author.

E-mail addresses: Zheng-Zhang@stu.ahu.edu.cn (Z. Zhang), e23301207@stu.ahu.edu.cn (Y. Yang), yanchen01@tyut.edu.cn (Y. Chen), jiangbb@hznu.edu.cn (B. Jiang), duliang@sxu.edu.cn (L. Du), zhoupeng@ahu.edu.cn (P. Zhou).¹ These authors contributed equally and should be regarded as co-first authors.

To tackle this problem, in this paper, we propose a new multi-view classification framework for out-of-distribution (OOD) generalization. In single-view classification, some works try to tackle the OOD problem by learning some invariant features across domains [14,15]. In OOD setting, the training data and test data often belong to different domains [16]. These conventional OOD methods divide features into causal features and non-causal features, and assume that the causal features remain invariant across multiple domains and the non-causal features may vary across different domains [17]. To this end, they apply several ways to discover the causal features, which are also called stable features. One of the popular ways is stable learning by reweighting samples [18,19]. In stable learning methods, they believe that the dependence between the causal and non-causal features is the key problem that prevents the models from generalizing to different domains, and thus they try to eliminate the dependencies. Specifically, these methods aim to learn a weight for each sample to minimize correlations between features, which eliminates feature dependencies [20]. However, since multi-view data have multiple views and each view contains various features, the dependencies between features are much more complicated than those in a single-view setting. In the multi-view setting, just imposing weights on samples may be insufficient to characterize the dependencies between features across all views. Intuitively, assume that there are N samples in the training set. Conventional stable learning methods need to learn N weights $w_1, \dots, w_N$, where w_i is the weight of the i th sample. Notice that the number of weights N is irrelevant to the number of views and features in each view. Therefore, if the views and features become increasingly complicated, the N weights will be insufficient to eliminate the dependencies between the features and views.

To address this issue, in our multi-view classification framework, we propose a novel Multi-view Stable Classification (MvSC) method. To tackle the OOD problem, we wish to learn stable representations for classification so that the model can achieve good performance across different domains. As discussed before, since the sample reweighting method in single-view stable learning may be insufficient to characterize the complicated dependencies, we design a new weighting mechanism, which is called the dual-weighting mechanism. The key idea is that, since we wish to obtain stable representations for multi-view data that eliminate the dependencies among dimensions of representations, we can directly impose weights on both the dimensions of representations and samples. In more detail, assuming that we have N samples and aim to learn d -dimensional representations for each dataset, we will learn N weights for samples and d weights for dimensions. Due to the d weights on the dimensions, we can control the dependencies between each dimension of representations across multiple views more directly and more precisely. Moreover, we further design a novel adaptive weight learning strategy that selectively updates sample or dimension weights based on feature representation correlations and the changing trends of sample or dimension weights. This adaptive weight learning during the training phase further enhances the model's generalization capabilities in complex environments.

The main contributions of this paper are summarized as follows:

- We propose a novel multi-view stable classification framework for OOD generalization. To the best of our knowledge, this is the first multi-view stable representation learning work to handle the general OOD problem.
- We propose an innovative dual-weighting mechanism that can precisely control the dependencies between representations. Moreover, we design a new adaptive weight learning strategy to effectively train the sample and dimension weights.
- Extensive experiments on OOD benchmarks demonstrate our method's superiority over state-of-the-art multi-view and OOD classification methods.

The remaining of this paper is organized as follows. Section 2 introduces some related work. Section 3 introduces our MvSC in detail.

Section 4 shows the experimental results. At last, Section 5 concludes this paper.

2. Related works

In this section, we introduce the related work of multi-view classification and stable learning briefly.

2.1. Multi-view classification

Multi-view learning aims to exploit the complementary information provided by multiple views while preserving their shared semantic information [21,22]. Multi-view classification applies this complementary and shared information to classify the data [23,24]. Early multi-view classification approaches typically integrate multiple views through fixed or learnable view-level weighting and fusion strategies [25,26]. For example, Zhang et al. [27] formulated multi-view representation learning from an information-theoretic perspective to balance cross-view consistency and complementarity. Pérez-Rúa et al. [25] introduced a general search space covering diverse multi-view fusion architectures.

Recent multi-view classification studies have increasingly focused on improving representation quality, information fusion, and robustness under complex data conditions. Representative efforts include prototype-based representation learning [1], cross-view inconsistency modeling [28], and deterministic multi-view information-bottleneck learning [29]. For example, Chao et al. [8] proposed TrashToTreasure, which decomposed multi-view information into useful and noisy components and further recovers potentially valuable information from the discarded components through information-theoretic interaction. Zou et al. [26] jointly modeled aleatoric, epistemic, and predictive uncertainties throughout representation learning and evidence-based fusion to obtain more reliable representations and clustering assignments. To address incomplete multi-view classification, Chao et al. [30] represented samples from different views as type-specific nodes in a heterogeneous graph, enabling missing views to be handled without explicit reconstruction while exploiting cross-view consistency and complementarity. The Cooperation of Experts framework [31] represented heterogeneous information using multiplex networks and coordinates view-specific and fused-network experts through a learnable confidence tensor and a large-margin mechanism. Meanwhile, uncertainty-aware methods dynamically estimate view reliability or predictive uncertainty to reduce the adverse effects of noisy and conflicting views during fusion [32–34].

Although the above methods can alleviate the effects of noisy, corrupted, or conflicting views, most of them are evaluated primarily under corruption-based distribution shifts. Such settings represent a relatively restricted form of OOD variation because the underlying domains and semantic distributions usually remain similar. In contrast, we consider a more general domain-generalization setting in which the target distribution is unavailable during training and may differ substantially from the source distributions. For example, a model may be trained using real photographs but evaluated using sketches. Existing multi-view classification methods rarely investigate this challenging setting, motivating the development of our stable multi-view representations that remain discriminative under previously unseen domain shifts.

2.2. Stable learning

Stable learning aims to identify predictive relationships that remain reliable under distribution shifts. A common strategy is sample reweighting, which adjusts the contribution of training samples to reduce undesirable correlations between predictive and spurious features [17,20]. Conventional approaches such as importance weighting commonly assume access to information about the target distribution and estimate density ratios between the training and target environments

[35]. Subsequent studies have investigated more challenging settings in which the target distribution is unknown during training [36].

Many stable-learning methods are motivated by causal invariance. For example, Shen et al. [37] introduced global confounder-balancing regularization to improve causal-feature identification and stable cross-domain estimation. Kuang et al. [38] employed autoencoders to learn stable low-dimensional representations in high-dimensional settings. Shen et al. [20] showed that reducing dependencies between relevant and irrelevant features improves prediction stability under distribution shifts. Zhang et al. [17] extended feature-decorrelation strategies to convolutional neural networks.

Recent studies have further investigated invariant representation learning and domain generalization under distribution shifts. Wang et al. [39] proposed a causally invariant feature discovery framework to identify predictive features that remained stable across environments. Chu et al. [14] developed a joint variational inference network that modeled domain-specific latent variables to improve generalization to unseen domains. Lin et al. [40] introduced an evolving intermediate domain to progressively bridge labeled and unlabeled source domains for semi-supervised domain generalization. Zhu et al. [16] proposed a path-information constraint that suppressed domain-specific information while preserving task-relevant information. However, unlike domain generalization, this setting permits model adaptation using target-domain samples during inference.

Another kind of work learns sample weights automatically. For example, Ren et al. [41] proposed a meta-learning algorithm that assigned sample weights according to their gradient directions. Zhou et al. [42] learned sample weights using bilevel optimization. He et al. [18] optimized robust weighted risks over distributional perturbations. Nevertheless, these stable-learning and domain-generalization methods are formulated and evaluated predominantly in single-view settings. How to jointly model view complementarity, view-dependent instability, and sample-level spurious correlations remains underexplored. Our method addresses this gap through a dual-weighting mechanism that simultaneously learns sample-level and feature-dimension-level importance for stable multi-view classification under unseen domain shifts.

3. Multi-view stable classification

In this section, we introduce our MvSC method.

3.1. Problem setting and overview framework

Given a multi-view training set $\mathcal{X} = \{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots, \mathbf{X}^{(V)}\}$ with V views and C classes, where each view $\mathbf{X}^{(i)} \in \mathbb{R}^{N \times d_i}$ contains N samples and d_i features, their labels set $\mathcal{Y} = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ where $\mathbf{y}_i$ is the one-hot vector of the ground-truth label for the i th sample, denoted as $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{iC}]$. $y_{ij} = 1$ if the i th sample belongs to the j th class, and 0 otherwise. In classification, we train a classifier on the training set $\mathcal{X}$ and $\mathcal{Y}$, and wish it can work well on the test set $\mathcal{X}_{te} = \{\mathbf{X}_{te}^{(1)}, \dots, \mathbf{X}_{te}^{(V)}\}$. Notice that in the OOD setting, the distribution of the training set $\mathcal{X}$ and $\mathcal{Y}$ may be very different from the distribution of the test set $\mathcal{X}_{te}$ and $\mathcal{Y}_{te}$.

To tackle this OOD problem in multi-view classification, we propose a novel stable multi-view classification network, whose architecture is shown in Fig. 1. For each view, the method applies a multi-layer perceptron (MLP) to extract the representations, and then applies a fusion method to obtain a consensus representation. To address the OOD issue, we design a dual-weighting mechanism (DWM) to learn stable representations via reweighting the samples and representations. At last, we obtain the final classification result from the stable representations. In the following, we will introduce each part of our method in more detail.

3.2. View-specific representation learning and fusion

For each view, we apply an independent MLP to extract a view-specific representation. For example, given the v th view, we use MLP

to extract representation $\mathbf{Z}^{(v)}$ from $\mathbf{X}^{(v)}$ as: $\mathbf{Z}^{(v)} = MLP_v(\mathbf{X}^{(v)})$, where $MLP_v(\cdot)$ denotes the MLP for the v th view, and $\mathbf{Z}^{(v)} \in \mathbb{R}^{N \times d_e}$ with d_e being the embedding dimension. Then, we apply a linear classification head, consisting of a linear layer and a Softmax activation function to obtain the classification result $\hat{\mathbf{y}}_i^{(v)}$ of the i th sample in the v th view:

$$\hat{\mathbf{y}}_i^{(v)} = \text{Softmax}(\mathbf{Z}_i^{(v)} \mathbf{W}^{(v)} + \mathbf{b}^{(v)}), \quad (1)$$

where $\mathbf{W}^{(v)} \in \mathbb{R}^{d_e \times C}$ and $\mathbf{b}^{(v)} \in \mathbb{R}^C$ are the weight matrix and bias vector of the classification head for view v .

After obtaining the prediction of the v th view, we use a cross-entropy loss to train the view-specific feature extraction network and classification head as:

$$\mathcal{L}^{(v)}(\mathbf{Z}^{(v)}, \mathbf{Y}) = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C -y_{ij} \log(\hat{y}_{ij}^{(v)}), \quad (2)$$

where y_{ij} and $\hat{y}_{ij}^{(v)}$ denotes the j th element of the ground truth $\mathbf{y}_i$ and predicted $\hat{\mathbf{y}}_i^{(v)}$, respectively. $\mathbf{Y} \in \mathbb{R}^{N \times C}$ is the ground-truth label matrix.

Notice that $\mathcal{L}^{(v)}(\mathbf{Z}^{(v)}, \mathbf{Y})$ is just a view-specific loss. To comprehensively make use of the information in multiple views, we need to ensemble the representations of different views. To this end, we learn a consensus representation $\mathbf{Z}$ with learnable weights γ_v as:

$$\mathbf{Z} = \sum_{v=1}^V \gamma_v \mathbf{Z}^{(v)}, \quad (3)$$

where the learnable weights γ_v satisfy $\sum_{v=1}^V \gamma_v = 1$ and $\gamma_v \geq 0$.

Similarly, we apply a linear classification head to the fused representation to obtain the consensus $\hat{\mathbf{y}}_i^{\text{fusion}}$:

$$\hat{\mathbf{y}}_i^{\text{fusion}} = \text{Softmax}(\mathbf{Z}_i \mathbf{W}^{\text{fusion}} + \mathbf{b}^{\text{fusion}}), \quad (4)$$

where $\mathbf{W}^{\text{fusion}} \in \mathbb{R}^{d_e \times C}$ and $\mathbf{b}^{\text{fusion}} \in \mathbb{R}^C$ are the weight matrix and bias vector of the consensus classification head, $\mathbf{Z}_i$ is the embedding of the i th sample.

Then we use a fusion loss $\mathcal{L}_{\text{fusion}}$, which is also a cross-entropy loss between the ground truth $\mathbf{Y}$ and the prediction $\hat{\mathbf{Y}}^{\text{fusion}}$ to train the network:

$$\mathcal{L}_{\text{fusion}}(\mathbf{Z}, \mathbf{Y}) = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C -y_{ij} \log(\hat{y}_{ij}^{\text{fusion}}). \quad (5)$$

Combining the view-specific loss and the fusion loss, we obtain the total loss:

$$\mathcal{L}(\mathbf{Z}, \mathbf{Y}) = \sum_{v=1}^V \mathcal{L}^{(v)}(\mathbf{Z}^{(v)}, \mathbf{Y}) + \mathcal{L}_{\text{fusion}}(\mathbf{Z}, \mathbf{Y}). \quad (6)$$

As introduced before, since the representations may contain irrelevant representations or confounders, it may limit the model trained by Eq. (6) from generalizing to new domains. Therefore, we modify Eq. (6) for OOD generalization by designing a new dual-weighting mechanism, which will be introduced in the next subsection.

3.3. Dual-weighting mechanism

The above mentioned multi-view representation learning assumes that the training and test data are i.i.d, which may be inappropriate to tackle the OOD problem. Stable learning offers a natural way to alleviate this problem by reducing unstable dependencies and spurious correlations in the learned representation, thereby encouraging the classifier to rely on relationships that remain more consistent across domains. However, directly apply stable learning to handle multi-view data may be inappropriate due to the complexity of multi-view data. Conventional stable learning tries to decorrelate the original features. Unfortunately, it is unreasonable to eliminate the dependencies between features in different views because they may be in different semantic spaces. To this end, we perform stable learning in the shared consensus space rather than in the original view-specific spaces. Moreover, because sample reweighting alone may be insufficient to characterize the complex dependencies in multi-view representations, we introduce a dual-weighting mechanism that jointly learns sample and dimension weights. With this stable learning, we effectively eliminate the confounders and spurious associations in the embedding, achieving better OOD generalization.

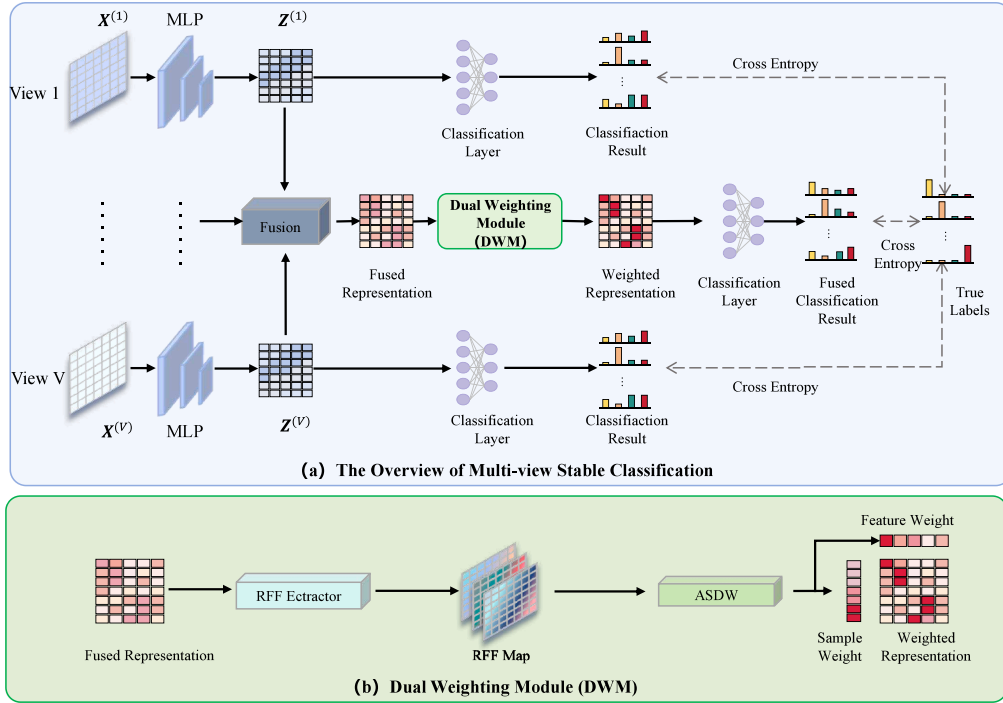


Fig. 1. The overall architecture of the proposed method. We apply an MLP to extract the representation of each view and ensemble them to obtain the consensus representation. To tackle the OOD problem, we design a novel dual-weighting mechanism (DWM), which is shown in the bottom block. DWM learns the sample weight and dimension weight to obtain a stable representation for final classification.

More specifically, the p th dimension and q th dimension of the learned consensus representation $\mathbf{Z} \in \mathbb{R}^{N \times d_e}$ may exhibit complex dependencies, which is the key problem to prevent the representations from being used in different domains [20]. Therefore, to obtain a stable representation matrix $\mathbf{Z}$, we need to eliminate the dependencies in it. We denote $\mathbf{Z}_{:,p}$ as the p th dimension of the fused representation $\mathbf{Z}$ (i.e., the p th column of matrix $\mathbf{Z}$), which captures certain semantic information in the unified representation space. The goal of the dual-weighting mechanism is to impose d_e weights on all dimensions of representation $\mathbf{Z}$ and N weights on all samples to eliminate the dependencies between any two representations $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$.

3.3.1. Evaluating representation dependencies

To eliminate the dependencies between $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$, we first need a way to estimate the dependencies between them. Notice that $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$ are learned from heterogeneous views, and thus they may contain complex non-linear dependencies. To characterize the non-linear dependencies, we apply the Hilbert-Schmidt Independence Criterion (HSIC) [43] to characterize the dependencies. In more detail, suppose $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$ are in space $\mathcal{Z}$, we try to find a kernel map $\phi: \mathcal{Z} \rightarrow \mathcal{H}$ where $\mathcal{H}$ is a Reproducing Kernel Hilbert Space (RKHS). From the probability perspective, we assume $\mathbf{Z}_{:,p}$ follows a distribution $P_{\mathbf{Z}_{:,p}}$, and the element $Z_{i,p}$ in $\mathbf{Z}_{:,p}$ can be seen as a variable sampled from distribution $P_{\mathbf{Z}_{:,p}}$. Then, given a map ϕ , we estimate the cross covariance matrix $\mathbf{C}_{pq}$ between $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$ on the RKHS $\mathcal{H}$ as follows:

$$\mathbf{C}_{pq} = \mathbb{E}((\phi(\mathbf{Z}_{:,p}) - \mathbb{E}(\phi(\mathbf{Z}_{:,p}))) \otimes (\phi(\mathbf{Z}_{:,q}) - \mathbb{E}(\phi(\mathbf{Z}_{:,q})))),$$

where $\otimes$ denotes the outer product. Then, the HSIC between $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$ can be defined as:

Definition 1. (HSIC) [43] Given an RKHS and a joint distribution $P_{\mathbf{Z}_{:,p}, \mathbf{Z}_{:,q}}$, we define the HSIC as the Hilbert-Schmidt norm of the cross covariance matrix $\mathbf{C}_{pq}$:

$$HSIC(\mathbf{Z}_{:,p}, \mathbf{Z}_{:,q}) = \|\mathbf{C}_{pq}\|_{HS}^2 \quad (7)$$

where $\|\cdot\|_{HS}$ denotes the Hilbert-Schmidt norm in RKHS.

In practice, the joint distribution $P_{\mathbf{Z}_{:,p}, \mathbf{Z}_{:,q}}$ is often unknown, and thus the expectation in HSIC is difficult to compute. To this end, we use the empirical version of HSIC as:

$$HSIC(\mathbf{Z}_{:,p}, \mathbf{Z}_{:,q}) = \|\mathbf{C}_{pq}\|_F^2 \quad (8)$$

where

$$\mathbf{C}_{pq} = \frac{1}{N-1} \sum_{i=1}^N \left[\left(\phi(Z_{i,p}) - \frac{1}{N} \sum_{j=1}^N \phi(Z_{j,p}) \right) \left(\phi(Z_{i,q}) - \frac{1}{N} \sum_{j=1}^N \phi(Z_{j,q}) \right)^T \right]. \quad (9)$$

From the definition of HSIC, we can find that HSIC can be used to measure the dependency of two variables, i.e., the less HSIC is, the more independent the two variables are. Now, to evaluate the dependencies by HSIC, we have one last thing to do, which is to determine the kernel map ϕ . Here, following [17], we also use the Random Fourier Feature (RFF) as the map function ϕ . In more detail, given an input z , $\phi(z)$ is a vector $\phi(z) = [\phi_1(z), \phi_2(z), \dots, \phi_b(z)]^T$, where b is the number of sampled functions. Each $\phi_i(z)$ is defined as:

$$\phi_i(z) = \sqrt{2} \cos(\omega z + \xi) \quad (10)$$

where ω is sampled from the standard Gaussian distribution, i.e., $\omega \sim N(0, 1)$ and ξ is sampled from a uniform distribution in the range $[0, 2\pi]$, i.e., $\xi \sim \text{Unif}(0, 2\pi)$.

3.3.2. Decorrelate by dual weighting mechanism

Equipped with the HSIC and RFF, we can evaluate the dependencies between any representations $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$. As introduced before, to decorrelate $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$, we wish to minimize the HSIC between them, i.e., we wish to minimize Eq. (8).

Traditional stable learning methods often impose a weight w_i on the i th sample and thus need to learn N weights $w_1, \dots, w_N$ to minimize the dependency. As introduced before, since $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$ come from multiple views and in each view there are various representations, only N weights may be insufficient to control such a complex dependency.

To tackle this issue, we propose a novel dual-weighting mechanism, which not only imposes weights on samples, but also imposes weights

Algorithm 1 Algorithm for adaptive sample and dimension weighting (ASDW).

Input: The initial weights $\mathbf{w}$ and α .

```

1: while not converged do
2:   Compute the gradient of  $\mathbf{w}$  and  $\alpha$  as  $\nabla \mathbf{w}$  and  $\nabla \alpha$ , respectively.
3:   if  $\|\nabla \mathbf{w}\|_2 / \text{length}(\mathbf{w}) \geq \|\nabla \alpha\|_2 / \text{length}(\alpha)$  then
4:     Update sample weights  $\mathbf{w} = \mathbf{w} - lr * \nabla \mathbf{w}$ .
5:     Normalize sample weights  $w_i = \frac{N * e^{w_i}}{\sum_{j=1}^N e^{w_j}}$ .
6:   else
7:     Update representation weights  $\alpha = \alpha - lr * \nabla \alpha$ .
8:     Normalize representation weights  $\alpha_i = \frac{d_e * e^{\alpha_i}}{\sum_{j=1}^{d_e} e^{\alpha_j}}$ .
9:   end if
10: end while

```

Output: Updated weights $\mathbf{w}$ and α .

on the dimensions of representations. In more detail, given the representation $\mathbf{Z}$, we try to learn two groups of weights. One group is N weights on samples, and the other group is d_e weights on the dimensions of representations. We denote the weights on samples as $\mathbf{w} = [w_1, \dots, w_N]$ and the weights on representations as $\alpha = [\alpha_1, \dots, \alpha_{d_e}]$.

Taking the weights into the cross covariance Eq. (9), we obtain the weighted cross covariance matrix as follows:

$$\mathbf{C}_{\mathbf{pq}; \mathbf{w}, \alpha} = \frac{1}{N-1} \sum_{i=1}^N \left[\left(w_i \alpha_p \phi(Z_{i,p}) - \frac{1}{N} \sum_{j=1}^N w_j \alpha_p \phi(Z_{j,p}) \right) \left(w_i \alpha_q \phi(Z_{i,q}) - \frac{1}{N} \sum_{j=1}^N w_j \alpha_q \phi(Z_{j,q}) \right)^T \right]. \quad (11)$$

Then we obtain the weighted HSIC as:

$$HSIC_{w\alpha}(\mathbf{Z}_{:,p}, \mathbf{Z}_{:,q}) = \|\mathbf{C}_{\mathbf{pq}; \mathbf{w}, \alpha}\|_F^2. \quad (12)$$

Therefore, we learn $\mathbf{w}$ and α by minimizing the weighted HSIC between all pairs of $\mathbf{Z}_{:,p}$ and $\mathbf{Z}_{:,q}$, i.e.,

$$\min_{\mathbf{w} \in \Delta_N, \alpha \in \Delta_{d_e}} \sum_{p < q} \|\mathbf{C}_{\mathbf{pq}; \mathbf{w}, \alpha}\|_F^2 + \|\mathbf{w}\|_2^2 + \|\alpha\|_2^2 \quad (13)$$

where $\Delta_N = \{\mathbf{w} \in \mathbb{R}_+^N \mid \sum_{i=1}^N w_i = N\}$ and $\Delta_{d_e} = \{\alpha \in \mathbb{R}_+^{d_e} \mid \sum_{i=1}^{d_e} \alpha_i = d_e\}$. On one hand, these constraints guarantee that the weights are bounded; on the other hand, since the summation of the weights is constant, they make the weights compete with each other, which leads to the sparsity of the weights. We also add the L2 regularization terms $\|\mathbf{w}\|_2^2$ and $\|\alpha\|_2^2$ to prevent overfitting and ensure numerical stability during optimization. The optimal $\mathbf{w}^*$ and α^* , which minimize the dependencies of the representations, make the representations more stable to handle the OOD problem.

Algorithm 2 Algorithm for multi-view stable classification (MvSC).

Input: Multi-view dataset $\mathbf{X}$.

```

1: Initialize  $\mathbf{w} = \mathbf{1}$  and  $\alpha = \mathbf{1}$ .
2: while not converged do
3:   for each batch do
4:     Compute  $\mathbf{Z}^{(v)}$  and  $\hat{y}_i^{(v)}$  for each view  $v$ .
5:     Compute fused representation  $\mathbf{Z}$ .
6:     Compute loss  $L_w$  and update parameters  $\theta$ .
7:   end for
8:   Optimize weights  $\mathbf{w}, \alpha$  according to Algorithm 1.
9: end while

```

Output: Network parameters θ , and weights $\mathbf{w}$ and α .

After obtaining the optimal weights $\mathbf{w}^*$ and α^* , we impose them on the classification loss functions Eq. (6), to obtain our final weighted

classification loss function:

$$\mathcal{L}_w = \sum_{i=1}^N w_i^* \left[L_{\text{fusion}}(\alpha^* \odot \mathbf{Z}_{\text{fused}, i}, y_i) + \sum_{v=1}^V L^{(v)}(Z_i^{(v)}, y_i) \right], \quad (14)$$

where $\odot$ is the element-wise production. We train the network by minimizing Eq. (14), aiming to learn a stable and dimension-independent multi-view representation, which can address the OOD problem for classification.

3.4. Training strategy

In our method, we have two groups of variables to be optimized. One group is the weights $\mathbf{w}$ and α , and the other group is the neural network parameters θ (e.g. $W^{(v)}$ and W). We first initialize all weights in $\mathbf{w}$ and α to 1, and then propose an iterative training strategy, where we train one group of variables while fixing the other group of variables. In more detail, we apply the SGD to optimize the parameters in the network of each view by minimizing Eq. (14). Then, we use the gradient descent method to optimize $\mathbf{w}$ and α . Since the two groups of variables are entangled in the objective function, we design an adaptive weight update method, called Adaptive Sample and Dimension Weighting (ASDW). We do not simultaneously update the two variables but decide which variable to update based on its gradient. Intuitively, we prioritize optimizing the variables with a larger gradient for effective descent. Since the length of $\mathbf{w}$ and α may differ, we normalize the gradient by dividing by the length of the variable. More specifically, we denote the gradient of $\mathbf{w}$ and α as $\nabla \mathbf{w}$ and $\nabla \alpha$, respectively. Then we compare $\|\nabla \mathbf{w}\|_2 / \text{length}(\mathbf{w})$ and $\|\nabla \alpha\|_2 / \text{length}(\alpha)$. If $\|\nabla \mathbf{w}\|_2 / \text{length}(\mathbf{w})$ is greater than $\|\nabla \alpha\|_2 / \text{length}(\alpha)$, we update $\mathbf{w}$ and otherwise we update α .

Notice that $\mathbf{w}$ and α have constraints, which may be violated during gradient descent. To tackle this problem, after the gradient descent, we apply Softmax on $\mathbf{w}$ and α to normalize them as:

$$w_i = \frac{N * e^{w_i}}{\sum_{j=1}^N e^{w_j}}, \quad \alpha_i = \frac{d_e * e^{\alpha_i}}{\sum_{j=1}^{d_e} e^{\alpha_j}} \quad (15)$$

By this way, $\mathbf{w}$ and α are projected to the Δ_N and Δ_{d_e} respectively. Algorithm 1 shows the pseudo-code of the process ASDW of updating $\mathbf{w}$ and α . The whole training process of our method is shown in Algorithm 2.

In the testing phase, we directly use the model with parameters θ for prediction without the weights $\mathbf{w}$ and α .

4. Experiments

In this section, we conduct extensive experiments on diverse scenarios to comprehensively evaluate our method.

4.1. Experimental setup and implementation details

We first introduce the datasets, compared methods, evaluation metrics, and implementation details, respectively.

4.1.1. Datasets

To demonstrate the effectiveness of handling the OOD problem, we conduct experiments on three benchmark OOD datasets, including two multi-view image datasets PACS [44] and VLCS [45], and a multi-view text dataset BBC [46]. PACS is a widely used dataset in domain generalization, comprising 7 object categories across four domains: Photo,

Table 1
Statistics of the datasets used in our experiments.

Dataset	#Samples	#features in each view	#Classes	Data Type
PACS	9991	512/512/1280/768	7	Image
VLCS	10,729	512/512/1280/768	5	Image
BBC	685	4659 / 4633 / 4665 / 4684	5	Text

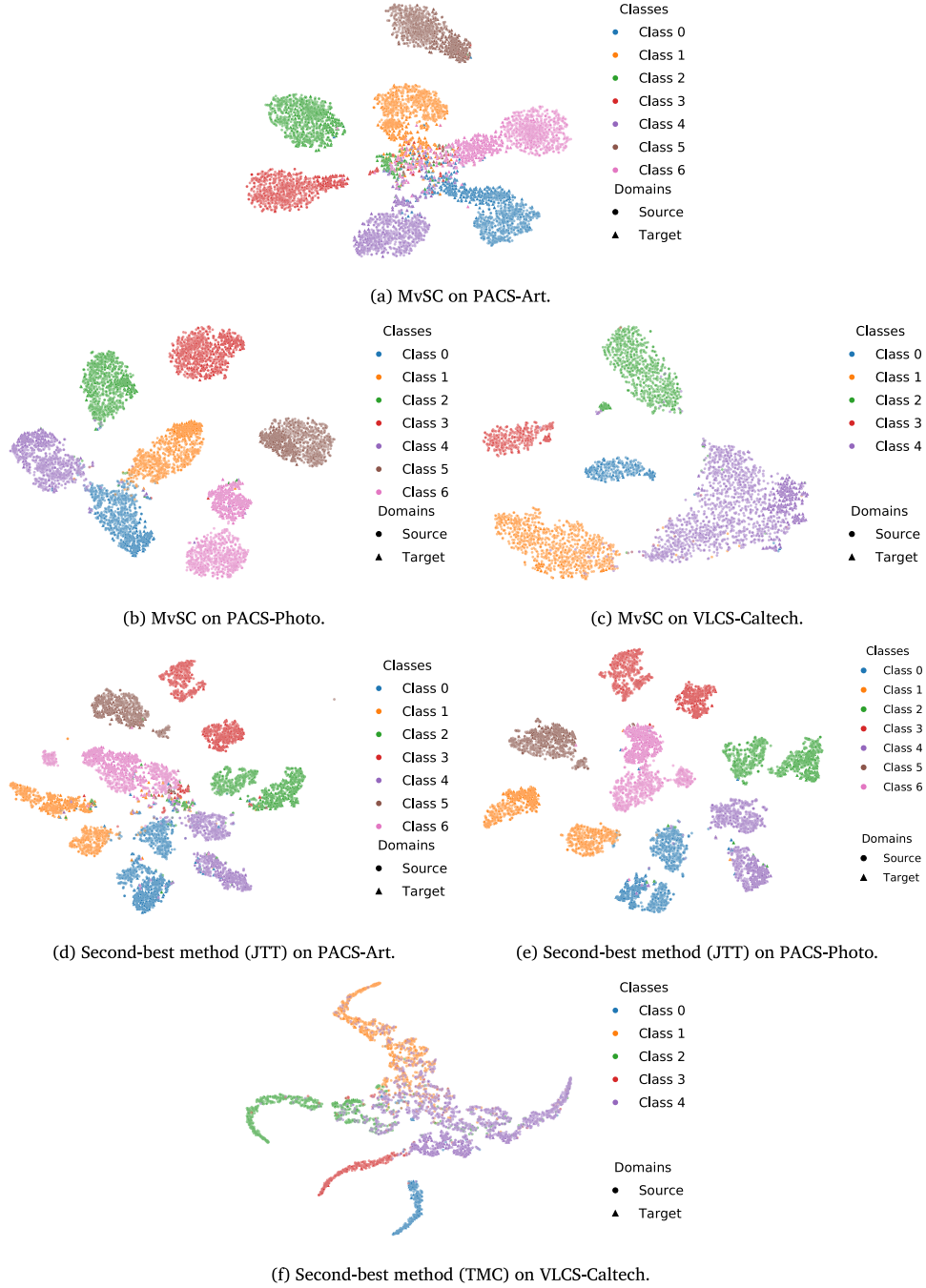


Fig. 2. t-SNE visualization of the representations learned by MvSC and the second-best methods. (a), (b), and (c) present the results of MvSC on PACS-Art, PACS-Photo, and VLCS-Caltech, respectively. (d), (e), and (f) present the results of the second-best methods on PACS-Art, PACS-Photo, and VLCS-Caltech, respectively. Samples with the same color belong to the same class, and circles and triangles denote source-domain and target-domain samples, respectively.

Art-painting, Cartoon, and Sketch. This dataset contains 9991 images, including 1670 photos, 2048 arts, 2344 cartoons, and 3929 sketches. VLCS is composed of five categories shared among 4 domains: Pascal, Labelme, Caltech, and Sun datasets. The VLCS dataset has 10,729 images, including 3376 images in Pascal, 2656 images in Labelme, 1415 images in Caltech, and 3282 images in Sun. Notice that both PACS and VLCS are single-view datasets. To evaluate the performance of multi-view methods on these datasets, we use pretrained ResNet-18 [47], VGG-16 [48], EfficientNet-B0 [49], and ViT-B/16 [50] to extract the features, respectively, and generate the multi-view data. The dimensions of the four views are 512, 512, 1280, and 768, respectively. Then, we follow the experimental protocol of [51] for both the datasets and utilize three do-

main as source domains for training and the remaining one as the target domain for test.

The BBC dataset [46] contains 685 news documents from five categories. Each document is represented by four views with dimensions of 4659, 4633, 4665, and 4684, respectively. To evaluate the OOD generalization, following [52], we apply a non-uniform split for it. Specifically, for the first three classes, we randomly assign 80% of samples to training, 10% to validation, and 10% to testing. For the remaining two classes, we assign 10% to training, 10% to validation, and 80% to testing. This creates a significant semantic drift between the training and test distributions.

Table 1 summarizes the datasets used in our experiments.

Table 2

Comparison with multi-view learning methods and out-of-distribution methods based on ACC and AUROC over PACS and VLCS. The best result in each column is highlighted in bold.

Method	Metric	PACS					VLCS				
		Art	Cartoon	Sketch	Photo	Avg.	labelme	caltech	pascal	sun	Avg.
TMC [34]	ACC	0.7749	0.5482	0.4413	0.8994	0.6660	0.5897	0.9906	0.7157	0.7462	0.7606
	AUROC	0.9649	0.9119	0.8684	0.9963	0.9354	0.7376	0.9993	0.8945	0.8553	0.8717
TMDLO [33]	ACC	0.7988	0.6621	0.4579	0.9641	0.7207	0.6863	0.9811	0.7720	0.7410	0.7951
	AUROC	0.9691	0.9410	0.8633	0.9984	0.9429	0.7655	0.9993	0.9298	0.8655	0.8900
ETMC [34]	ACC	0.8179	0.6788	0.5108	0.9623	0.7424	0.6650	0.9693	0.7256	0.7360	0.7740
	AUROC	0.9663	0.9167	0.8180	0.9972	0.9245	0.7239	0.9826	0.8297	0.8507	0.8467
DMVLS [6]	ACC	0.7343	0.6182	0.4622	0.9736	0.6971	0.6499	0.9835	0.7522	0.7137	0.7748
	AUROC	0.9392	0.8729	0.7498	0.9976	0.8899	0.7070	0.9993	0.9038	0.7802	0.8476
RCML [53]	ACC	0.8104	0.6711	0.4737	0.9660	0.7303	0.6901	0.9882	0.7542	0.7147	0.7868
	AUROC	0.9771	0.9446	0.8623	0.9961	0.9450	0.7782	0.9991	0.9272	0.8713	0.8940
CSMVIB [29]	ACC	0.7769	0.6625	0.5253	0.8994	0.7160	0.6562	0.9575	0.7108	0.7198	0.7611
	AUROC	0.9599	0.9213	0.8688	0.9901	0.9350	0.7210	0.9979	0.9151	0.8674	0.8754
TTT [8]	ACC	0.8042	0.6216	0.5452	0.9485	0.7298	0.6562	0.9835	0.7907	0.7228	0.7883
	AUROC	0.9580	0.9222	0.8392	0.9871	0.9335	0.7443	0.9970	0.9290	0.7997	0.8675
TMLC [23]	ACC	0.7954	0.5879	0.4972	0.9677	0.7121	0.6311	0.9906	0.7927	0.7472	0.7904
	AUROC	0.9707	0.9353	0.8556	0.9981	0.9399	0.8143	0.9921	0.9388	0.8369	0.8955
ERM [54]	ACC	0.8145	0.6566	0.5582	0.9670	0.7491	0.6637	0.9670	0.7453	0.7371	0.7783
	AUROC	0.9626	0.9309	0.8494	0.9975	0.9351	0.7775	0.9993	0.8405	0.8601	0.8694
RWY [55]	ACC	0.8086	0.6369	0.5772	0.9660	0.7472	0.6562	0.9858	0.7947	0.7463	0.7957
	AUROC	0.9724	0.9120	0.8589	0.9977	0.9353	0.7936	0.9991	0.9423	0.8688	0.9009
JTT [15]	ACC	0.8164	0.6566	0.5411	0.9760	0.7475	0.6637	0.9788	0.7532	0.7157	0.7779
	AUROC	0.9723	0.9309	0.8308	0.9980	0.9330	0.7775	0.9926	0.9153	0.8483	0.8834
StableNet [17]	ACC	0.8008	0.6988	0.5479	0.9353	0.7457	0.6299	0.9245	0.7522	0.6772	0.7460
	AUROC	0.8906	0.8458	0.7661	0.9697	0.8680	0.7048	0.9672	0.8309	0.7449	0.8119
MAPLE [42]	ACC	0.8101	0.6502	0.5073	0.9623	0.7325	0.6223	0.9829	0.7897	0.7391	0.7835
	AUROC	0.9781	0.9405	0.8738	0.9980	0.9476	0.7979	0.9998	0.9329	0.8862	0.9042
MARW [18]	ACC	0.5259	0.4424	0.3276	0.8701	0.5415	0.6754	0.9413	0.5953	0.7289	0.7352
	AUROC	0.9218	0.9094	0.8727	0.9855	0.9224	0.7654	0.9759	0.9163	0.8324	0.8725
MvSC(Ours)	ACC	0.8242	0.6455	0.5798	0.9868	0.7591	0.6373	0.9906	0.8094	0.7533	0.7976
	AUROC	0.9787	0.9357	0.8894	0.9995	0.9508	0.8088	0.9998	0.9508	0.8717	0.9078

4.1.2. Compared methods and evaluation metrics

To systematically evaluate the proposed model's effectiveness in addressing OOD problem in multi-view classification, we compare with eight state-of-the-art multi-view classification methods, including TMC [34], TMDLO [33], ETMC [34], RCML [53], CSMVIB [29], DMVLS [6], TTT [8], and TMLC [23]; and six state-of-the-art single-view OOD classification methods, including ERM [54], RWY [55], JTT [15], StableNet [17], MAPLE [42], and MARW [18].

When running the compared methods, we use the codes and parameter settings provided by their authors. For single-view methods that cannot directly handle multi-view data, we concatenate all views to form a single-view dataset and feed it into the single-view methods. We use two widely-used evaluation metrics, including Classification Accuracy (ACC) and the Area Under the Receiver Operating Characteristic Curve (AUROC), to evaluate the classification performance.

4.1.3. Implementation details

All view-specific encoders in our method share the same architecture. Each encoder first applies Layer Normalization to the input features, followed by a three-layer fully connected network with dimensions input $\rightarrow 256 \rightarrow 128 \rightarrow 100$, where the first two layers use ReLU activation and dropout, while the final layer has no activation function. The model is trained for 30 epochs with a batch size of 256. We optimize the backbone parameters θ using AdamW with an initial learning rate of 10^{-3} , weight decay of 10^{-5} , and a cosine annealing learning-rate schedule. The sample weights w and dimension weights α are optimized using SGD optimizer with a learning rate of 10^{-4} and momentum of 0.9.

For HSIC estimation, we employ RFF to approximate the Gaussian kernel. The number of random Fourier features is set to 8 in all experiments. The random projection weights are sampled from a standard normal distribution, and the phase offsets are sampled uniformly from $[0, 2\pi]$. The same HSIC estimation configuration is adopted for all

datasets. We also fix the random seed throughout the experiments to ensure reproducibility.

4.2. Experimental results

Table 2 shows the results of all methods on PACS and VLCS datasets. Since PACS and VLCS have 4 domains, there are 4 scenarios in these datasets, where in each scenario, we use one domain as the test domain and the other three domains as the training domain. The second row of Table 2 shows the test domain. For example, "Art" means we use the "Art" domain as the test domain and the other three domains as the training domain. "Avg." denotes the average result of all 4 scenarios. Table 2 shows that our method outperforms other methods across most scenarios, and achieve the best performance on average for both datasets. When compared with multi-view methods, since we especially design a dual-weighting mechanism to tackle the OOD problem, our method performs better when the test domain is different from the training domain. It well demonstrates our motivation that the conventional multi-view classification may suffer from the OOD problem. When compared with OOD classification methods, ours also often achieves better performance. On one hand, conventional OOD classification methods are designed for single-view data, which may be inappropriate to handle multi-view data; and on the other hand, conventional OOD methods only impose one kind of weight on the samples, which may not be enough to decorrelate the dependencies among representations. Compared with these methods, our dual-weighting mechanism can characterize and eliminate the dependencies better, leading to more stable representations for OOD classification.

Table 3 shows the results on the BBC dataset. The results are similar. Our method can achieve best performance compared to all multi-view and OOD methods. Besides, OOD methods often perform better than traditional multi-view methods. It shows that the existing OOD methods are better handling the distribution shifting, which means the test

Table 3
Comparison with multi-view learning and out-of-distribution methods on the BBC dataset. The best result is highlighted in bold.

Metric	Multi-view Classification Methods						OOD Classification Methods						Ours		
	TMC [34]	TMDLO [33]	ETMC [34]	DMVLS [6]	RCML [53]	CSMVIB [29]	TTT [8]	TMLC [23]	ERM [54]	RWY [55]	JTT [15]	StableNet [17]	MAPLE [42]	MARW [18]	MvSC
ACC	0.6519	0.2410	0.7156	0.6401	0.5226	0.4681	0.7698	0.3515	0.7747	0.7064	0.7248	0.7544	0.8052	0.4628	0.9205
AUROC	0.9591	0.8658	0.9735	0.9237	0.9636	0.9772	0.9912	0.8560	0.9728	0.9807	0.9817	0.9425	0.9803	0.9901	0.9987

Table 4

Efficiency results of all methods on the PACS-Art dataset.

Method	#Paras (M)	FLOPs (M)	Time _{tr} (s)	Time _{inf} (s)
TMC	0.022	0.044	38.15	0.14
TMDLO	0.022	0.043	37.21	0.14
ETMC	1.728	3.455	181.19	0.45
DMVLS	1.210	2.544	68.90	0.76
RCML	0.022	0.043	67.88	0.10
CSMVIB	5.080	10.208	200.71	0.13
TTT	2.136	4.274	713.028	0.23
TMLC	0.0215	0.043	68.203	0.06
ERM	1.972	1.970	30.98	0.22
RWY	1.972	1.970	30.50	0.26
JTT	1.972	1.970	29.38	0.21
StableNet	0.974	0.972	61.02	0.13
MAPLE	1.708	3.411	21.61	0.19
MARW	0.394	0.788	144.16	0.14
MvSC (Ours)	0.970	1.945	57.35	0.13

domain and training domain are the same but the distribution of each class is significantly different. In this setting, our method also outperforms the conventional OOD methods, demonstrating the effectiveness and superiority of our method.

4.3. Efficiency results

To evaluate the efficiency and scalability of different methods, we report the number of parameters, FLOPs, training time, and inference time of all methods on PACS-Art dataset in Table 4. MvSC maintains a compact model size with only 0.970M parameters and shows competitive training and inference efficiency compared with existing methods. Compared with most baselines, MvSC achieves a better balance between model complexity and computational cost, indicating that it is lightweight and practical for multi-view out-of-distribution generalization.

Although MARW has a smaller model size and some methods have slightly lower training cost, their classification performance is inferior to MvSC according to the ACC results. In contrast, MvSC achieves the best overall classification accuracy while keeping the parameter size and inference time moderate. These results demonstrate that MvSC provides a favorable trade-off between effectiveness and efficiency.

4.4. Ablation study

We conduct some ablation studies to evaluate the effectiveness of our dual-weighting mechanism in Table 5. We first compare the dual-weighting method (simultaneously using $\mathbf{w}$ and α) with the single weighting method, i.e., the sample weighting (only using $\mathbf{w}$) and dimension weighting (only using α). From Table 5, we can see that, compared to vanilla method, using the sample weights or using the dimension weights can both improve the performance, demonstrating the effectiveness of both the weighting mechanisms. Moreover, using both weights further performs better than using single weights, well demonstrating the superiority of our dual-weighting mechanism compared with the conventional single-weighting method.

Besides, we also conduct experiments to show the effect of our adaptive sample and dimension weighting method. The fourth row in Table 5 shows the results of our method that simultaneously updates $\mathbf{w}$ and α , and the fifth row shows our method with ASDW (i.e., using Algorithm 1). It can be seen that the adaptive updating performs better than the conventional simultaneous updating, demonstrating the effectiveness of Algorithm 1.

Moreover, to further evaluate the effects of our dual-weighting mechanism, we conduct an ablation study to show whether it can decorrelate the representations. Specifically, we compute the average pairwise HSIC values of the learned representations before and after applying the learned dual weights. A lower HSIC value indicates weaker statistical

Table 5
Ablation study of our method evaluated by ACC.

Components			PACS					VLCS				
w	α	Adaptive Update	Art	Cartoon	Sketch	Photo	Avg.	Labelme	Caltech	Pascal	Sun	Avg.
×	×	×	0.8164	0.6117	0.5316	0.9814	0.7359	0.6010	0.9764	0.7936	0.7401	0.7778
✓	×	×	0.8208	0.6399	0.5424	0.9844	0.7469	0.6235	0.9788	0.8015	0.7442	0.7870
×	✓	×	0.8218	0.6390	0.5472	0.9850	0.7483	0.6275	0.9787	0.8005	0.7401	0.7867
✓	✓	×	0.8218	0.6418	0.5530	0.9826	0.7498	0.6311	0.9835	0.8036	0.7523	0.7926
✓	✓	✓	0.8242	0.6455	0.5798	0.9868	0.7591	0.6373	0.9906	0.8094	0.7533	0.7976

Table 6
Average pairwise HSIC values with and without the proposed dual-weighting mechanism. Lower values indicate weaker statistical dependencies.

Setting	PACS-Art	PACS-Cartoon	PACS-Sketch	PACS-Photo	VLCS-LabelMe	VLCS-Caltech	VLCS-Pascal	VLCS-Sun
Without dual weighting	0.3937	0.4655	0.7220	0.3585	0.8251	0.7019	0.4703	0.6146
With dual weighting	0.3802	0.4495	0.5619	0.3471	0.8097	0.6486	0.4267	0.5557

Table 7
Effect of the number of random Fourier features on the PACS-Art dataset.

Number of RFF features	5	8	20	50
ACC	0.8154	0.8242	0.8184	0.8169
AUROC	0.9784	0.9787	0.9792	0.9785
Dual-weighting time (s)	18.55	20.46	27.10	43.64
Total training time (s)	57.90	59.15	70.61	80.04
Dual-weighting time ratio	32.04%	34.58%	38.39%	54.57%

dependence among representation dimensions. The results are shown in Table 6. It can be seen that the proposed dual-weighting mechanism consistently reduces HSIC. For example, on PACS-Art, the HSIC value decreases from 0.3937 to 0.3802; on PACS-Sketch, it decreases from 0.7220 to 0.5619; and on VLCS-Pascal, it decreases from 0.4703 to 0.4267. These results provide direct evidence that our method effectively suppresses statistical dependencies in the learned representations, which is consistent with the motivation of reducing spurious correlations for out-of-distribution generalization.

We further investigate the influence of the number of random Fourier features on both classification performance and computational efficiency. Specifically, we vary the RFF dimension among {5, 8, 20, 50} and conduct experiments on the PACS-Art dataset. We report ACC, AUROC, the running time of the dual-weighting module, the total training time, and the proportion of the dual-weighting time to the total training time. The results are shown in Table 7. We can see that increasing the RFF dimension from 5 to 8 improves ACC from 0.8154 to 0.8242 and AUROC from 0.9784 to 0.9787. When the dimension is increased to 20, AUROC further improves slightly to 0.9792, whereas ACC decreases to 0.8184. Meanwhile, the running time of the dual-weighting module increases from 20.46 s to 27.10 s. Using 50 RFF features does not yield further performance improvements but substantially increases the dual-weighting time to 43.64 s, accounting for 54.57% of the total training time. Overall, using 8 RFF features achieves the highest ACC, a nearly optimal AUROC, and the lowest total training time among the evaluated settings. Therefore, in our implementation, we set the number of random Fourier features to 8 in all experiments to achieve a favorable balance between classification performance and computational efficiency.

4.5. t-SNE visualization

To intuitively evaluate the discriminability and cross-domain stability of the learned representations, we employ t-SNE to visualize the fused features from the source and target domains. Before applying t-SNE, we first extract the learned fused representations from the trained model and standardize them using z-score normalization. We then perform t-SNE with perplexity set to 30, PCA initialization, a fixed random seed of 2026, and the default learning rate setting. The results are shown in

Fig. 2. Different colors denote different classes, and circles and triangles represent source-domain and target-domain samples, respectively. We show the t-SNE results of our method on PACS-Art, PACS-Photo, and VLCS-Caltech in Fig. 2(a)–(c), respectively. Moreover, we also show the results of the second-best methods in Fig. 2(d)–(f).

On PACS-Art and PACS-Photo, the second-best method, i.e., JTT, often maps the source domain and target domain samples to different groups, e.g. the red class, green class, and blue class. In our method, we map the source domain and the target domain samples together, generating more compact intra-class groups. These observations demonstrate that the proposed dual-weighting mechanism effectively suppresses spurious correlations and learns stable and discriminative representations, thereby improving generalization to unseen domains. On VLCS-Caltech, compared to the second-best method, i.e., TMC, our method provides a clearer inter-class separation, demonstrating that our method can learn more discriminative representations for classification.

5. Conclusion

This paper focused on the challenging problem of OOD generalization in multi-view classification. We proposed the first deep stable multi-view classification method specifically designed for general OOD scenarios. In the method, we designed a novel dual-weighting mechanism, which explicitly controls representation dependencies to eliminate spurious correlations, thereby learning a stable representation for multi-view data. In the dual-weighting mechanism, we learned the sample weights and dimension weights and carefully designed an adaptive weighting method to effectively train the model. Extensive experiments on OOD benchmarks demonstrated our method's superiority in generalizing to different domains.

CRedit authorship contribution statement

Zheng Zhang: Writing – original draft, Visualization, Software, Methodology, Data curation; **Yongxi Yang:** Writing – original draft, Validation, Methodology, Investigation, Data curation; **Yan Chen:** Writing – review & editing, Supervision, Methodology; **Bingbing Jiang:** Writing – review & editing, Supervision; **Liang Du:** Writing – review & editing, Supervision; **Peng Zhou:** Writing – review & editing, Supervision, Project administration, Methodology, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China grants 62576002 and 62176001, and the Natural Science Project of Anhui Provincial Education Department grant 2023AH030004. This work is also supported by the Anhui Provincial Key Research and Development Program under grant 202304a05020047.

Data availability

Data will be made available on request.

References

- [1] S. Ma, Z. Dong, R. Cong, S. Kwong, X. Shao, Prototype-based multi-view fine-grained 3D classification and ad-hoc interpretability, *Pattern Recognit.* 179 (2026) 113596–113608.
- [2] S. HU, J. WANG, J. FAN, S. LI, J. QIN, X. JIANG, P. LV, M. XU, Relational contrastive multi-view clustering, *Front. Comput. Sci.* 21 (2) 2102320.
- [3] X. Liu, L. Liu, Q. Liao, S. Wang, Y. Zhang, W. Tu, C. Tang, J. Liu, E. Zhu, One pass late fusion multi-view clustering, in: *Proc. Int. Conf. Mach. Learn.*, Vol. 139, PMLR, 2021, pp. 6850–6859.
- [4] J. Xu, M. Meng, J. Liu, J. Wu, Deep multi-view clustering with diverse and discriminative feature learning, *Pattern Recognit.* 161 (2025) 111322–111333.
- [5] Z. Lou, Y. Wu, K. Zhang, C. Zhang, B. Ji, S. Hu, Structure-enhanced self-supervised weighted information bottleneck for multiview clustering, *IEEE Trans. Neural Netw. Learn. Syst.* 37 (2026) 3727–3740.
- [6] X. Wan, J. Liu, X. Liu, Y. Wen, H. Yu, S. Wang, S. Yu, T. Wan, J. Wang, E. Zhu, Decouple then classify: a dynamic multi-view labeling strategy with shared and specific information, in: *Proc. Int. Conf. Mach. Learn.*, Vol. 235, PMLR, 2024, pp. 49941–49956.
- [7] X. Yang, H. Che, M.-F. Leung, Tensor-based unsupervised feature selection for error-robust handling of unbalanced incomplete multi-view data, *Inf. Fusion* 114 (2025) 102693–102716.
- [8] G. Chao, M. Zhang, X. Wang, J. Wen, W. Ding, D. Chu, TrashToTreasure: an informative and interactive multi-view classification framework, *IEEE Trans. Knowl. Data Eng.* 38 (2026) 3264–3276.
- [9] J.-H. Yang, Y. Zhou, L. Zhang, H.-C. Li, Mixed-noise robust tensor multi-view clustering via adaptive dictionary learning, *Inf. Fusion* 123 (2025) 103322.
- [10] P. Zhou, L. Du, Learnable graph filter for multi-view clustering, in: *Proc. ACM Int. Conf. Multimedia*, ACM, 2023, pp. 3089–3098.
- [11] Y. Bengio, T. Deleu, N. Rahaman, N.R. Ke, S. Lachapelle, O. Bilaniuk, A. Goyal, C.J. Pal, A meta-transfer objective for learning to disentangle causal mechanisms, in: *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–14.
- [12] B. Recht, R. Roelofs, L. Schmidt, V. Shankar, Do imagenet classifiers generalize to imagenet?, in: *Proc. Int. Conf. Mach. Learn.*, Vol. 97, PMLR, 2019, pp. 5389–5400.
- [13] P. Zhou, R. Li, Z. Ling, L. Du, X. Liu, Fair clustering ensemble with equal cluster capacity, *IEEE Trans. Pattern Anal. Mach. Intell.* 47 (3) (2025) 1729–1746.
- [14] J.-Z. Chu, B. Pan, X. Xu, T.-Y. Shi, Z.-W. Shi, T. Li, Joint variational inference network for domain generalization, *Pattern Recognit.* 154 (2024) 110587–110596.
- [15] E.Z. Liu, B. Haghighi, A.S. Chen, A. Raghunathan, P.W. Koh, S. Sagawa, P. Liang, C. Finn, Just train twice: improving group robustness without training group information, in: *Proc. Int. Conf. Mach. Learn.*, Vol. 139, PMLR, 2021, pp. 6781–6792.
- [16] J. Zhu, J. Zhuo, S. Wang, PIC: domain generalization by path information constraint, *Pattern Recognit.* 168 (2025) 111769.
- [17] X. Zhang, P. Cui, R. Xu, L. Zhou, Y. He, Z. Shen, Deep stable learning for out-of-distribution generalization, in: *CVPR*, 2021.
- [18] Y. He, P. Tian, R. Xu, X. Shen, X. Zhang, P. Cui, Model-agnostic random weighting for out-of-distribution generalization, in: *Proc. ACM SIGKDD Int. Conf. KDD*, 12, 2024, pp. 1050–1061.
- [19] Z. Zhang, P. Zhou, A. Yao, L. Du, X. Liu, Interpretable subspace clustering, *IEEE Trans. Pattern Anal. Mach. Intell.* 48 (5) (2026) 6020–6034.
- [20] Z. Shen, P. Cui, T. Zhang, K. Kunag, Stable learning via sample reweighting, *Proc. AAAI Conf. Artif. Intell.* 34 (04) (2020) 5692–5699.
- [21] J. Wen, C. Liu, S. Deng, Y. Liu, L. Fei, K. Yan, Y. Xu, Deep double incomplete multi-view multi-label learning with incomplete labels and missing views, *IEEE Trans. Neural Networks Learn. Syst.* 35 (8) (2024) 11396–11408.
- [22] C. Liu, J. Wen, Z. Wu, X. Luo, C. Huang, Y. Xu, Information recovery-driven deep incomplete multiview clustering network, *IEEE Trans. Neural Networks Learn. Syst.* 35 (11) (2024) 15442–15452.
- [23] C. Tang, Y. Shi, G. Lin, L. Xing, L. Shi, Trusted multi-view learning for long-tailed classification, in: *Proc. AAAI Conf. Artif. Intell.*, Vol. 40, 2026, pp. 25814–25822.
- [24] S. Hu, J. Fan, Y. Wu, J. Wang, X. Jiang, P. Lv, M. Xu, To the best of trust: full-stage trusted multi-modal clustering, *IEEE Trans. Image Process.* 35, (2026), 8635–8647.
- [25] J.-M. Pérez-Rúa, V. Vielzeuf, S. Pateux, M. Baccouche, F. Jurie, MFAS: multimodal fusion architecture search, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6966–6975.
- [26] G. Zou, S. Hu, T. Chen, Y. Ye, Stop pushing friends away: trustworthy hybrid contrastive multi-modal clustering, *Inf. Fusion* 136, (2026) 104565.
- [27] H. Zhang, V.M. Patel, R. Chellappa, Hierarchical multimodal metric learning for multimodal classification, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3057–3065.
- [28] X. Yue, Z. Dong, Y. Chen, S. Xie, Evidential dissonance measure in robust multi-view classification to resist adversarial attack, *Inf. Fusion* 113 (2025) 102605–102618.
- [29] Q. Zhang, M. Lu, J. Xin, B. Chen, Towards a robust multi-view information bottleneck using Cauchy–Schwarz divergence, *Inf. Fusion* 118 (2025) 102934–102945.
- [30] G. Chao, J. Qi, J. Li, B. Lee, D. Chu, Incomplete multi-view classification via graph neural network on heterogeneous graph, *Inf. Fusion* 121 (2025) 103137.
- [31] S. Wang, S. Huang, J. Yuan, Z. Shen, Z. Kang, Cooperation of experts: fusing heterogeneous information with large margin, in: *Proc. Int. Conf. Mach. Learn.*, 2025, pp. 63169–63185.
- [32] Y. Geng, Z. Han, C. Zhang, Q. Hu, Uncertainty-aware multi-view representation learning, *Proc. AAAI Conf. Artif. Intell.* (2022) 7545–7553.
- [33] W. Liu, X. Yue, Y. Chen, T. Denooux, Trusted multi-view deep learning with opinion aggregation, in: *Proc. AAAI Conf. Artif. Intell.*, Vol. 36, 2022, pp. 7585–7593.
- [34] Z. Han, C. Zhang, H. Fu, J.T. Zhou, Trusted multi-view classification with dynamic evidential fusion, *IEEE Trans. Pattern Anal. Mach. Intell.* 45 (2) (2022) 2551–2566.
- [35] T. Fang, N. Lu, G. Niu, M. Sugiyama, Rethinking importance weighting for deep learning under distribution shift, *Adv. Neural Inf. Process. Syst.* 33 (2020) 11996–12007.
- [36] J. Liu, Z. Shen, Y. He, X. Zhang, R. Xu, H. Yu, P. Cui, Towards out-of-distribution generalization: a survey, (2021). *arXiv preprint arXiv:2108.13624*
- [37] Z. Shen, P. Cui, K. Kuang, B. Li, P. Chen, Causally regularized learning with agnostic data selection bias, in: *Proc. ACM Int. Conf. Multimedia*, 2018.
- [38] K. Kuang, P. Cui, S. Athey, R. Xiong, B. Li, Stable prediction across unknown environments, in: *Proc. ACM SIGKDD Conf. Knowl. Discov. Data Min.*, 2018.
- [39] Y. Wang, K. Yu, G. Xiang, F. Cao, J. Liang, Discovering causally invariant features for out-of-distribution generalization, *Pattern Recognit.* 150 (2024) 110338–110350.
- [40] L. Lin, H. Xie, Z. Sun, W. Chen, W. Liu, Y. Yu, L. Zhang, Semi-supervised domain generalization with evolving intermediate domain, *Pattern Recognit.* 149 (2024) 110280–110294.
- [41] M. Ren, W. Zeng, B. Yang, R. Urtasun, Learning to reweight examples for robust deep learning, in: *Proc. Int. Conf. Mach. Learn.* 2018, pp. 4334–4343.
- [42] X. Zhou, Y. Lin, R. Pi, W. Zhang, R. Xu, P. Cui, T. Zhang, Model agnostic sample reweighting for out-of-distribution learning, in: *Proc. Int. Conf. Mach. Learn.*, PMLR, 2022, pp. 27203–27221.
- [43] A. Gretton, O. Bousquet, A. Smola, B. Schölkopf, Measuring statistical dependence with Hilbert–Schmidt norms, in: *International Conference on Algorithmic Learning Theory*, Springer, 2005, pp. 63–77.
- [44] D. Li, Y. Yang, Y.-Z. Song, T.M. Hospedales, Deeper, broader and artier domain generalization, in: *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2017, pp. 5542–5550.
- [45] A. Torralba, A.A. Efros, Unbiased look at dataset bias, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2011.
- [46] D. Greene, P. Cunningham, Practical solutions to the problem of diagonal dominance in kernel document clustering, in: *Proc. Int. Conf. Mach. Learn.*, 2006, pp. 377–384.
- [47] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [48] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: *Proc. Int. Conf. Learn. Represent.*, 2015.
- [49] M. Tan, Q.V. Le, EfficientNet: rethinking model scaling for convolutional neural networks, in: *Proc. Int. Conf. Learn. Represent.*, Vol. 97 of *Proceedings of Machine Learning Research*, PMLR, 2019, pp. 6105–6114.
- [50] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: transformers for image recognition at scale, in: *Proc. Int. Conf. Learn. Represent.*, 2021.
- [51] F.M. Carlucci, A. D’Innocente, S. Bucci, B. Caputo, T. Tommasi, Domain generalization by solving jigsaw puzzles, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2229–2238.
- [52] U. Arora, W. Huang, H. He, Types of out-of-distribution texts and how to detect them, in: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 10687–10701.
- [53] C. Xu, J. Si, Z. Guan, W. Zhao, Y. Wu, X. Gao, Reliable conflictive multi-view learning, in: *Proc. AAAI Conf. Artif. Intell.*, Vol. 38, 2024, pp. 16129–16137.
- [54] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer science & business media, 2013.
- [55] B.Y. Idrissi, M. Arjovsky, M. Pezeshki, D. Lopez-Paz, Simple data balancing achieves competitive worst-group-accuracy, in: *Proc. Conf. Causal Learn. Reason.*, PMLR, 2022, pp. 336–351.