IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 21 June 2023, accepted 3 July 2023, date of publication 13 July 2023, date of current version 9 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3294984

RESEARCH ARTICLE

DAC: Disentanglement-and-Calibration Module for Cross-Domain Few-Shot Classification

HAO ZHENG[1], QIANG ZHANG[2], AND ASAKO KANEZAKI[1], (Member, IEEE)
[1]Tokyo Institute of Technology, Meguro City, Tokyo 152-8550, Japan
[2]The Hong Kong University of Science and Technology (Guangzhou), Nansha, Guangzhou 511400, China

Corresponding author: Asako Kanezaki (kanezaki@c.titech.ac.jp)

ABSTRACT Cross-domain few-shot classification (CD-FSC) aims to develop few-shot classification models trained on seen domains but tested on unseen domains. However, the cross-domain setup poses a challenge in the form of domain shift between the training and testing domains. Previous research has demonstrated that the encoder can disentangle features into domain-shared and domain-specific features. However, poorly estimated domain-specific features can lead to inadequate generalization on the unseen domain. This paper proposes a disentanglement-and-calibration module (DAC) to address this issue. The disentanglement component separates the features into domain-shared and domain-specific features, while the calibration component corrects the domain-specific features. We demonstrate that the DAC module can significantly enhance the generalization capability of several baseline methods. Furthermore, we show that MatchingNet with the DAC module outperforms existing state-of-the-art methods by 10%-11% when trained on mini-ImageNet, CUB-200, Cars196, Places365 and tested on Plantae dataset.

INDEX TERMS Cross-domain few-shot classification, disentanglement, domain shift, representation learning.

I. INTRODUCTION

Few-shot classification (FSC) aims at classifying novel classes with the support of limited labeled examples. Many effective algorithms have been proposed to solve the FSC problem, such as MatchingNet [1], ProtoNet [2], GNNNet [3], TabLLm [4] and CAD [5]. However, in reality, the assumption that both the source domain and target domain are sampled from the same dataset [6] is often not applicable, as there is usually a domain gap between the two datasets. The presence of a domain gap has inspired research into cross-domain few-shot classification (CD-FSC) [6], [7], [8], [9], [10], [11]. CD-FSC involves training a model to extract recognizable features from one or more domains and testing it on an unseen domain. Domain generalization is a task similar to CD-FSC but without the limitation of data quantity. The differences between these tasks are vividly illustrated in Figure 1.

The associate editor coordinating the review of this manuscript and approving it for publication was Turgay Celik.

In recent years, there has been an increasing amount of research focused on CD-FSC. One approach that has garnered attention within the research community is LFT [7], which involves inserting linear transformation layers into the encoder. This method aligns features from different domains by mapping them into the same space and simulating the domain shift. However, it is unclear if it is necessary to transfer all features to bridge the domain gap. In other words, for cross-domain problems, it may be more appropriate to transfer only *domain-specific* features.

Previous methods [8], [9], [10] have demonstrated that features can be disentangled into *domain-specific* features, which are unique to each domain, and *domain-shared* features, which are independent of the domain. Domain-shared features exhibit common characteristics, regardless of whether the inputs are from a seen or unseen domain, while domain-specific features represent the private components of each domain. Both domain-shared features and domain-specific features contain discriminative information that can benefit image classification.

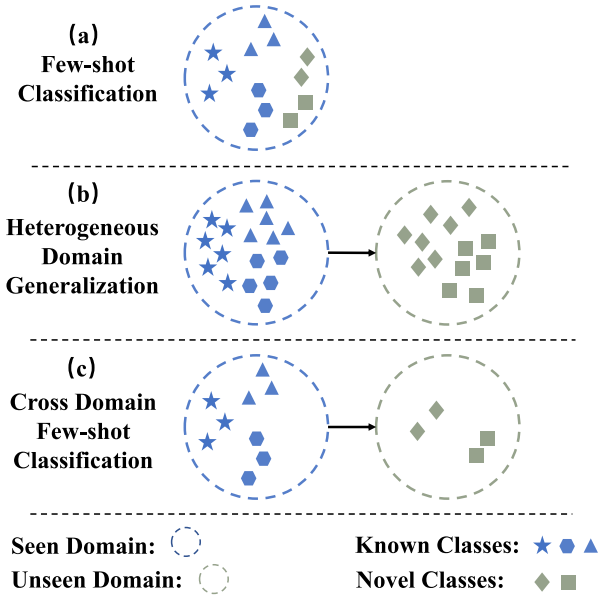

FIGURE 1. Comparison of few-shot learning, heterogeneous domain generalization, and cross-domain few-shot learning.

Typically, a model can accurately extract its domain-shared component, even when an image from an unseen domain not present during the training stage is sampled. However, the model may fail to estimate the domain-specific component of the image; Poor estimation of domain-specific components can result in the poor estimation of discriminative information and failure of generalization. Meta-FDMixUp [9] eliminates domain-specific parts to mitigate the negative impact of poor estimation.

In this paper, we utilize the learning-to-learn ability of meta-learning to simulate the seen-to-unseen shift through a disentanglement-and-calibration process. Specifically, in each meta task, we disentangle the features from two domains into domain-shared and domain-specific features and then calibrate the domain-specific features of one domain to better adapt to the domain shift. The key to our method lies in the design of a module to solve the CD-FSC problem, which we call Disentanglement-And-Calibration (DAC).

When designing the disentanglement module, we propose three fundamental principles. Firstly, the distributions of domain-shared features should be consistent across different domains. Secondly, the distributions of domain-specific features should vary significantly across different domains. Lastly, the distributions of domain-shared and domain-specific features should also exhibit diversity. In the case of cross-domain disentanglement, the DADA framework [10] introduces mutual information to quantitatively evaluate the differences between distributions, which necessitates an additional mutual information estimation module. To simplify the problem, we design a disentanglement module that enforces a Gaussian distribution for the output, which facilitates the computation of the discrepancy between different feature groups. In terms of the calibration module, we implicitly model the bias between predicted and actual domain-specific features. Based on this model, we find that the Residual Block is highly suitable for mitigating the bias.

Experimental results have conclusively demonstrated the criticality of the position of the DAC module in the encoder. A shallow insertion of the DAC module may impede the disentanglement module from identifying shared information across diverse domains, as complex texture information may be difficult to discern. Conversely, only domain-specific semantic information may remain if the DAC module is placed too deep. Therefore, numerous experiments have been conducted to determine the optimal position for the DAC module.

Our contributions can be summarized into three key aspects:

1) We introduce the DAC module to validate our hypothesis. The disentanglement component partitions features into Gaussian-distributed domain-shared and domain-specific features, and the calibration component rectifies the discrepancy between predicted and real domain-specific features.
2) Through extensive experiments and comparison among all possible positions, we identify the optimal location for inserting the DAC module.
3) We perform numerous experiments to demonstrate the efficacy of our approach. In most scenarios, our method surpasses previous state-of-the-art methods by a significant margin.

II. RELATED WORK

A. FEW-SHOT CLASSIFICATION

Few-shot classification aims to train a classifier that can recognize new categories with a limited number of labeled examples during the training stage. This is demonstrated in Figure 1(*a*). Numerous noteworthy achievements have been proposed in the field of few-shot classification, with many of them utilizing meta-learning approaches. These approaches can be broadly categorized into three processing views. The first processing view is based on "learning to fine-tune" [12], [13], [14], [15], [16]. This approach involves learning the optimal ways to fine-tune a classifier so that it can effectively generalize to novel classes with minimal cost. Another perspective is "learning to compare," which not only requires the model to identify similarities and differences between images but also to find the metric distance. This enables the FSC model to better adapt to new, unseen input images. Geometric ideas such as cosine similarity and Euclidean distance are utilized in [1] and [17], respectively. In [18], CNN models are introduced, and in [3], graph neural networks are adopted to distinguish between images. The third perspective is "learning to augment," which involves hallucinating unseen new classes to enhance the generalization ability of models with limited samples [19], [20].

However, according to Chen et al. [6], these methods may experience performance degradation when the distribution of

extracted image features differs greatly in different domains of the task. To address this challenge, Tseng et al. [7] proposed cross-domain few-shot learning, and BSCD-FSL [21] established a new benchmark.

Existing methods based on meta-learning aim to fill the domain shift by transferring features or using ensemble learning. Some methods address the CD-FSC problem by adjusting feature distributions, such as in [7], [9] and [22]. Multiple encoders are integrated to recognize novel classes in [11] and [23].

Some methods, such as [7], [24], and [25], use only source data. STARTUP [26] relaxes the setting by allowing the model to access many unlabeled target data during training, while Meta-FDMixup [9] resorts to a setting where only a few labeled target images are available. ATA [27] proposes the task augmentation method which can generate the inductive bias-adaptive challenging tasks. AFA [28] simulates distribution variations by maximizing the domain discrepancy.

B. DOMAIN GENERALIZATION

Domain generalization (DG) method endeavors to achieve better results on unseen domains without touching any instances from unseen domains during the training stage [29]. Zhou et al. [30] give a comprehensive literature review about vision DG. Unlike other related problems such as domain adaptation, DG considers scenarios where the target data are inaccessible during model learning. Depending on the category to be recognized whether novel or not, DG can be divided into homogeneous DG and heterogeneous DG. Most of the CD-FSC methods comply with heterogeneous DG, which is shown in Figure 1(b).

There have been significant breakthroughs in domain generalization in recent years. The majority of existing DG approaches attempt to learn domain-invariant features across source domains [31], [32], [33], [34], [35], [36]. The motivation behind this is straightforward: domain-invariant features are expected to be robust to any unseen domain shift. Disentanglement-based methods [29], [31], [33] decouple features and apply domain-invariant features to different downstream tasks. Niu et al. [37] attempt to improve classifiers from known domains. Adversarial learning has been introduced in [38] and [39] for data augmentation. Recent breakthroughs in meta-learning have inspired researchers to utilize the learning-to-learn strategy to significantly enhance model generalization. Li et al. [40] were the first to apply meta-learning to DG. The majority of existing meta-learning-based DG methods [41], [42], [43], [44], [45], [46] follow their learning paradigm.

C. CROSS-DOMAIN DISENTANGLEMENT

Liu et al. [47] define a disentangled representation as a latent feature that is sensitive to changes in one factor while being unresponsive to other factors. They found that decoupling work is highly beneficial for representation learning. Beta-VAE [48] designs a classifier-based metric that compares the disentangled features learned by different models. Thus, independent hidden variable factors can be automatically discovered, avoiding the need for prior knowledge.

Chen et al. [49] proposed InfoGAN, which uses the lower bound of mutual information as the optimization objective. The objective is to maximize the mutual information between the hidden variables and a small set of observations, which can be used to separate and identify specific features.

In [50], Hwang et al. leverage information-theoretic principles to achieve a decoupled representation of cross-domain images using Variational Autoencoder (VAE).

Decoupled representations have shown to outperform the original features in cross-domain tasks, especially when the number of available training samples is limited. Our approach involves specialized decoupling analysis and tuning for each unique domain. We utilize convolutional neural networks to obtain the average value and variance value of the features and ensure they follow a Gaussian distribution.

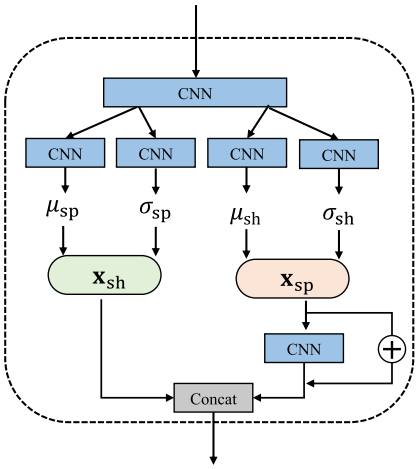

FIGURE 2. The inner structure of proposed DAC structure.

III. METHODOLOGY

A. PRELIMINARY

Firstly, we define $\mathcal{D}^{\text{seen}} = \{D_1^{\text{seen}}, D_2^{\text{seen}}, \ldots, D_N^{\text{seen}}\}$ as a collection of N seen domains. Each domain is comprised of a set of data-label pairs, denoted as $D_i^{\text{seen}} = \{X_i, Y_i\}$, where X_i and Y_i represent the images and their corresponding labels, respectively.

During the testing stage, the target domain is denoted as D^{unseen}, which is mutually exclusive with any domain in $\mathcal{D}^{\text{seen}}$. D^{unseen} is utilized to evaluate the model's classification capability on novel classes.

B. DISENTANGLEMENT-AND-CALIBRATION MODULE

As mentioned in Section I, poor estimation of domain-specific components can result in suboptimal discrimination and limited generalization. To address this issue, we propose the Disentanglement-and-Calibration (DAC) module, which decouples image features into domain-shared and

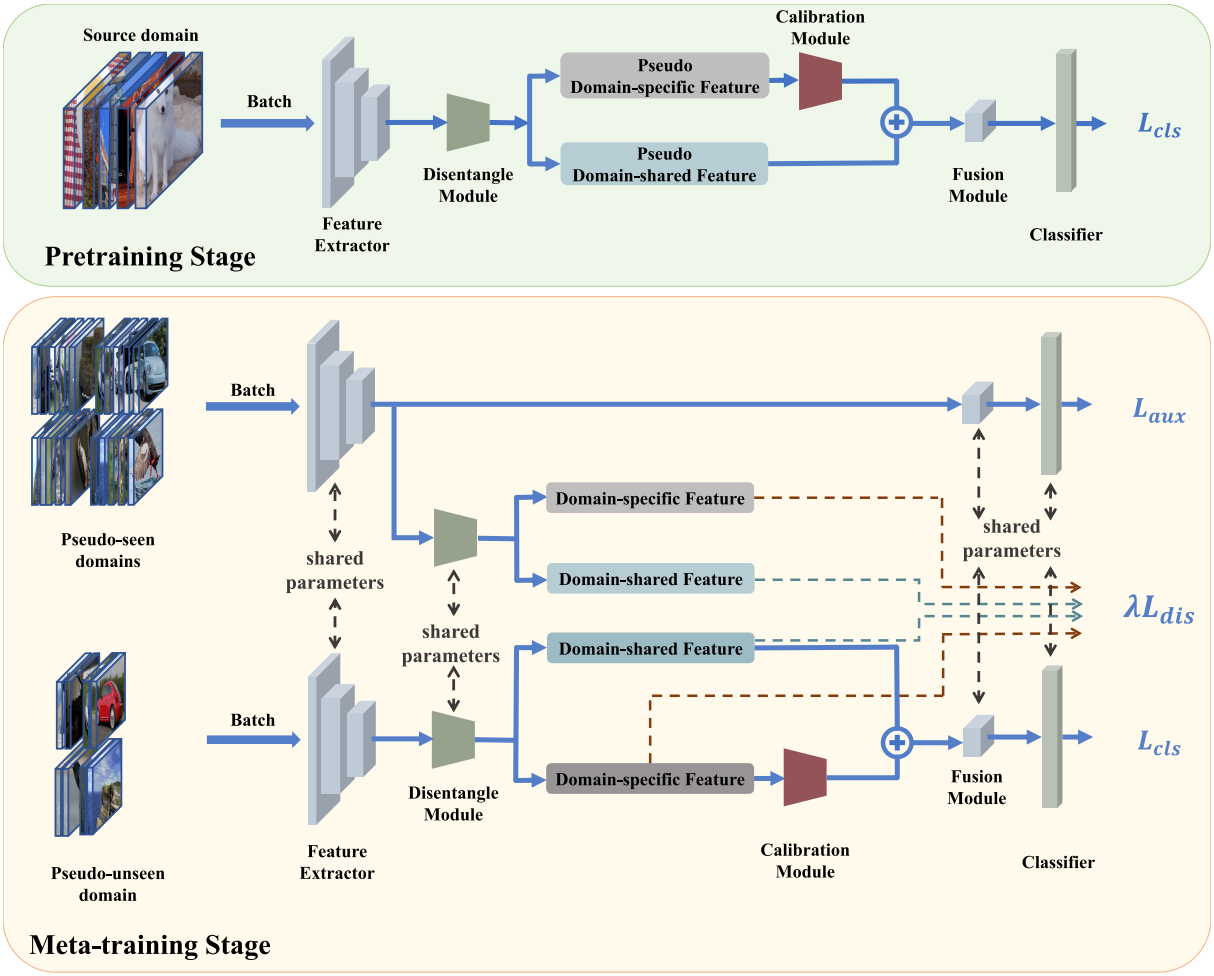

FIGURE 3. In the training stage, we first pretrain the entire network model using mini-ImageNet as the source domain. In the subsequent meta-training stage, we divide the available datasets into pseudo-seen and pseudo-unseen domains within each meta task. We then use our training strategy to simulate the generalization process from seen to unseen domains.

domain-specific components and then calibrates the estimated bias of the domain-specific components. Technologically, we utilize the reparameterization trick [51] as the foundation for feature disentanglement. We use the notations $\mathbf{x}_{\text{sh}}$ and $\mathbf{x}_{\text{sp}}$ to represent domain-shared and domain-specific features, respectively. As illustrated in Figure 2, we employ VAE-like modules to learn the average value $\boldsymbol{\mu}$ and variation value $\boldsymbol{\sigma}$. We then use these to construct $\mathbf{x}_{\text{sh}}$ and $\mathbf{x}_{\text{sp}}$ as follows:

$$\begin{aligned}\mathbf{x}_{\text{sp}} &= \mathbf{z} \cdot \exp\left(\boldsymbol{\sigma}_{\text{sp}}\right) + \boldsymbol{\mu}_{\text{sp}} \\ \mathbf{x}_{\text{sh}} &= \mathbf{z} \cdot \exp\left(\boldsymbol{\sigma}_{\text{sh}}\right) + \boldsymbol{\mu}_{\text{sh}},\end{aligned} \tag{1}$$

where $\mathbf{z}$ represents the a random noise vector sampled from a standard normal distribution. The subscripts sh and sp represent domain-shared and domain-specific components, respectively.

For disentanglement, we hope that the domain-shared features of different domain images are close to each other while the domain-specific features are pushed away from each other. At the same time, the domain-shared and domain-specific features of the same domain images should be separated. Thus, the loss function $\mathcal{L}_{\text{dis}}$ for the disentanglement task is:

$$\begin{aligned}\mathcal{L}_{\text{dis}} = {} & d(\mathbf{x}^i_{\text{sh}}, \mathbf{x}^j_{\text{sh}}) + \max(\eta - d(\mathbf{x}^i_{\text{sh}}, \mathbf{x}^i_{\text{sp}}), 0) \\ & + \max(\eta - d(\mathbf{x}^j_{\text{sh}}, \mathbf{x}^j_{\text{sp}}), 0) \\ & + \max(\eta - d(\mathbf{x}^i_{\text{sp}}, \mathbf{x}^j_{\text{sp}}), 0),\end{aligned} \tag{2}$$

where superscripts i and j ($i \neq j$) are used to distinguish the domain to which the features belong. $d(\mathbf{x}^i_{\text{sh}}, \mathbf{x}^j_{\text{sh}})$ measures the discrepancy between $\mathbf{x}^i_{\text{sh}}$ and $\mathbf{x}^j_{\text{sh}}$ by:

$$d(\mathbf{x}^i_{\text{sh}}, \mathbf{x}^j_{\text{sh}}) = \sqrt{\left\|\boldsymbol{\mu}^i_{\text{sh}} - \boldsymbol{\mu}^j_{\text{sh}}\right\|_2^2 + \left\|\boldsymbol{\sigma}^i_{\text{sh}} - \boldsymbol{\sigma}^j_{\text{sh}}\right\|_2^2}. \tag{3}$$

After passing through the disentanglement module, the features in the encoder can be represented discretely as either domain-shared features or domain-specific features. As previously discussed, the inadequate learning of domain-specific features is the cause of poor generalization, whereas the domain-shared features can be seen as the common part learned by each domain. Therefore, only the domain-specific

features require calibration, which is precisely what our calibration module is designed for.

There is always an error $\varepsilon = \mathcal{E}(\mathbf{x}_{\text{sp}}, \hat{\mathbf{x}}_{\text{sp}})$ between the model's estimated domain-specific features $\hat{\mathbf{x}}_{\text{sp}}$ and the actual domain-specific features $\mathbf{x}_{\text{sp}}$. It is evident that explicitly modeling this error ε is challenging because the features are located in a high-dimensional latent space. Nevertheless, it is fortuitous that such errors can be efficiently computed and corrected using the residual module. The operation of this module can be described as follows:

$$\begin{aligned}\mathbf{x}_{\text{sp}} &= \hat{\mathbf{x}}_{\text{sp}} + \varepsilon \\ &= \hat{\mathbf{x}}_{\text{sp}} + \mathcal{E}(\mathbf{x}_{\text{sp}}, \hat{\mathbf{x}}_{\text{sp}}) \\ &= \hat{\mathbf{x}}_{\text{sp}} + \hat{\mathcal{E}}(\hat{\mathbf{x}}_{\text{sp}}),\end{aligned} \tag{4}$$

where $\hat{\mathcal{E}}$ represents the estimated error between $\mathbf{x}_{\text{sp}}$ and $\hat{\mathbf{x}}_{\text{sp}}$. This formula highlights the same operational principle as that of the residual block, thus substantiating the rationality of the residual module.

C. NETWORK TRAINING STRATEGY

Our training strategy consists of two stages: pre-training and meta-training, which is illustrated in Figure 3. During the pre-training stage, the network is trained on mini-ImageNet, where the cross-entropy loss is minimized to update the network's basic classification recognition capability.

During the meta-training stage, we randomly select two datasets as the pseudo-seen domain and pseudo-unseen domain within each meta task. Initially, we train the vanilla classification task on the pseudo-seen domain. The network is updated by minimizing the cross-entropy loss $\mathcal{L}_{\text{aux}}$ to enhance its categorization ability on the *pseudo-seen* domain. $\mathcal{L}_{\text{aux}}$ is defined by:

$$\mathcal{L}_{\text{aux}} = -\frac{1}{B}\sum_{i=1}^{B}\sum_{j=1}^{C} y_{i,j}\log \hat{y}_{i,j}, \tag{5}$$

$$\mathcal{L} = \mathcal{L}_{\text{aux}} + \mathcal{L}_{\text{cls}} + \lambda\mathcal{L}_{\text{dis}}, \tag{6}$$

where λ represents a hyperparameter that balances the relative importance of the classification and disentanglement tasks in the overall loss function $\mathcal{L}$.

IV. RESULTS

A. EXPERIMENTAL SETUP

1) DATASETS

To validate the effectiveness of the proposed method, we employ five widely used datasets in the CD-FSC research community, namely mini-ImageNet [13], CUB-200 [52], Cars196 [53], Places365 [54], and Plantae [55]. These datasets are sampled from the real world, exhibiting significant domain gaps among them. Mini-ImageNet dataset comprises non-overlapping categories split into three parts: *train/val/test*. We utilize these parts as *base/val/novel* datasets, respectively, to train/validate/test few-shot classification models. Regarding the CUB-200 dataset, we randomly select categories with a ratio of 2 : 1 : 1 as *base/val/novel* datasets. The same operation is carried out on the remaining datasets.

2) BASELINES AND COMPETITORS

We compare our proposed DAC method with four baseline few-shot classification methods, namely MatchingNet [1], ProtoNet [2], RelationNet [18], and GNNNet [3]. Since our feature-wise transformation method aims to extract more generalizable features using a single feature extractor, we do not consider comparing ensemble learning methods that use multiple encoders. Additionally, we choose two feature-wise transformation methods, LFT [7], LRP [25], ATA [27] and AFA [28] as competitors. We quote the performance reported in these original papers to ensure the authority of the results. We also conduct experiments that the competitors ignored. For a fair comparison, we use their released code and default parameters in these additional experiments.

3) IMPLEMENTATION DETAILS

We employ ResNet-10 as the feature extractor, which is commonly used in CD-FSC methods for baselines and competitors. In the pre-training stage, we train the feature extractor, disentanglement module, and calibration module on mini-ImageNet with a 64-category classification task. The whole model is trained for 400 epochs with a batch size of 16. We utilize a leave-one-out setting where we select one unseen domain from CUB-200, Cars196, Places365, and Plantae as the test dataset. The remaining domains and the mini-ImageNet domain are used as seen domains for training. In the meta-training stage, we randomly select two available datasets as primary and auxiliary domains within each epoch. We use Adam optimizer with an initial learning rate of 0.001 and train each stage for 400 epochs. The weight of disentanglement loss L_{dis} is set to 0.001. Finally, we report the average classification accuracy of 1,000 episodes in the test stage. All of these basic settings are the same as those used in the baselines and competitors.

4) TESTING STAGE

During the testing stage of our experiment, we perform meta tasks, where we randomly select C categories from the unseen domain D^{unseen}, and sample K images from each category to build the support set. Our experiment includes two settings: 5-way 1-shot and 5-way 5-shot. To test the classification accuracy, we choose 16 images from each selected category and feed them into the FSC models, which are equipped with our DAC module. We repeat this process for 1,000 episodes and report the average classification accuracy as the final result.

B. RESULTS OF GENERALIZATION FROM MULTI-SOURCE DOMAINS

The term ''multi-source domains'' refers to the ability of a model to access multiple datasets during the training stage. In our experiment, we pre-trained our model on the mini-ImageNet dataset and selected one unseen domain from the

TABLE 1. The results of few-shot classification obtained by training with different plug-and-play modules on different datasets. The symbol "w/" indicates that the baseline is inserted by the following method. The best results are highlighted in bold. The symbol $*$ denotes that no related paper has conducted this group of experiments, so we re-implemented it with their released code.

			CUB-200	Cars196	Places365	Plantae
5-way 1-Shot	MatchingNet [1]	-	$37.90 \pm 0.55\%$	$28.96 \pm 0.45\%$	$49.01 \pm 0.65\%$	$33.21 \pm 0.51\%$
		w/ LFT [7]	$43.29 \pm 0.59\%$	$30.62 \pm 0.48\%$	$52.51 \pm 0.67\%$	$35.12 \pm 0.54\%$
		w/ ATA [27]	$39.65 \pm 0.40\%$	$32.22 \pm 0.40\%$	$53.63 \pm 0.50\%$	$36.42 \pm 0.40\%$
		w/ AFA [28]	$41.02 \pm 0.40\%$	$\mathbf{33.52 \pm 0.40\%}$	$\mathbf{54.66 \pm 0.50\%}$	$37.60 \pm 0.40\%$
		w/ DAC (Ours)	$\mathbf{44.53 \pm 0.60\%}$	$33.03 \pm 0.51\%$	$49.95 \pm 0.65\%$	$\mathbf{43.21 \pm 0.63\%}$
	ProtoNet [2]	-*	$39.71 \pm 0.54\%$	$31.18 \pm 0.48\%$	$40.68 \pm 0.60\%$	$34.55 \pm 0.54\%$
		w/ LFT* [7]	$31.76 \pm 0.49\%$	$28.26 \pm 0.45\%$	$40.29 \pm 0.61\%$	$29.21 \pm 0.46\%$
		w/ DAC (Ours)	$\mathbf{42.67 \pm 0.59\%}$	$\mathbf{32.53 \pm 0.50\%}$	$\mathbf{41.88 \pm 0.62\%}$	$\mathbf{35.40 \pm 0.58\%}$
	RelationNet [18]	-	$44.33 \pm 0.59\%$	$29.53 \pm 0.45\%$	$47.76 \pm 0.63\%$	$33.76 \pm 0.52\%$
		w/ LFT [7]	$48.38 \pm 0.63\%$	$32.21 \pm 0.51\%$	$\mathbf{50.74 \pm 0.66\%}$	$35.00 \pm 0.52\%$
		w/ LRP [25]	$45.64 \pm 0.42\%$	$30.00 \pm 0.32\%$	$48.74 \pm 0.45\%$	$36.04 \pm 0.38\%$
		w/ DAC (Ours)	$\mathbf{49.10 \pm 0.63\%}$	$\mathbf{32.54 \pm 0.55\%}$	$48.98 \pm 0.67\%$	$\mathbf{38.25 \pm 0.59\%}$
	GNNNet [3]	-	$49.46 \pm 0.73\%$	$32.95 \pm 0.56\%$	$51.39 \pm 0.80\%$	$37.15 \pm 0.60\%$
		w/ LFT [7]	$51.51 \pm 0.80\%$	$34.12 \pm 0.63\%$	$\mathbf{56.31 \pm 0.80\%}$	$42.09 \pm 0.68\%$
		w/ LRP* [25]	$49.90 \pm 0.70\%$	$32.27 \pm 0.53\%$	$51.72 \pm 0.76\%$	$39.68 \pm 0.63\%$
		w/ ATA [27]	$45.00 \pm 0.50\%$	$33.61 \pm 0.40\%$	$53.57 \pm 0.50\%$	$34.42 \pm 0.40\%$
		w/ AFA [28]	$46.86 \pm 0.50\%$	$34.25 \pm 0.40\%$	$54.04 \pm 0.60\%$	$36.76 \pm 0.40\%$
		w/ DAC (Ours)	$\mathbf{53.06 \pm 0.75\%}$	$\mathbf{35.34 \pm 0.56\%}$	$52.96 \pm 0.82\%$	$\mathbf{43.51 \pm 0.69\%}$
5-way 5-Shot	MatchingNet [1]	-	$51.92 \pm 0.80\%$	$39.87 \pm 0.51\%$	$61.82 \pm 0.57\%$	$47.29 \pm 0.51\%$
		w/ LFT [7]	$61.41 \pm 0.57\%$	$43.08 \pm 0.55\%$	$64.99 \pm 0.59\%$	$48.32 \pm 0.57\%$
		w/ ATA [27]	$57.53 \pm 0.40\%$	$45.73 \pm 0.40\%$	$67.87 \pm 0.40\%$	$51.05 \pm 0.40\%$
		w/ AFA [28]	$59.46 \pm 0.40\%$	$46.13 \pm 0.40\%$	$\mathbf{68.87 \pm 0.40\%}$	$52.43 \pm 0.40\%$
		w/ DAC (Ours)	$\mathbf{62.38 \pm 0.56\%}$	$\mathbf{46.38 \pm 0.59\%}$	$65.67 \pm 0.55\%$	$\mathbf{58.66 \pm 0.57\%}$
	w/ ProtoNet [2]	-*	$57.47 \pm 0.55\%$	$42.03 \pm 0.55\%$	$58.37 \pm 0.58\%$	$\mathbf{48.53 \pm 0.55\%}$
		w/ LFT* [7]	$51.65 \pm 0.53\%$	$42.22 \pm 0.55\%$	$\mathbf{60.40 \pm 0.60\%}$	$44.61 \pm 0.60\%$
		w/ DAC (Ours)	$\mathbf{60.33 \pm 0.58\%}$	$\mathbf{43.03 \pm 0.56\%}$	$56.85 \pm 0.56\%$	$45.69 \pm 0.58\%$
	RelationNet [18]	-	$62.13 \pm 0.74\%$	$40.64 \pm 0.54\%$	$64.34 \pm 0.57\%$	$46.29 \pm 0.56\%$
		w/ LFT [7]	$64.99 \pm 0.54\%$	$43.44 \pm 0.59\%$	$\mathbf{67.35 \pm 0.54\%}$	$50.39 \pm 0.52\%$
		w/ LRP [25]	$62.71 \pm 0.39\%$	$41.05 \pm 0.37\%$	$66.08 \pm 0.40\%$	$48.78 \pm 0.37\%$
		w/ DAC (Ours)	$\mathbf{65.96 \pm 0.55\%}$	$\mathbf{43.71 \pm 0.56\%}$	$64.08 \pm 0.56\%$	$\mathbf{52.75 \pm 0.57\%}$
	GNNNet [3]	-	$69.26 \pm 0.68\%$	$48.91 \pm 0.67\%$	$72.59 \pm 0.67\%$	$58.36 \pm 0.68\%$
		w/ LFT [7]	$73.11 \pm 0.68\%$	$49.88 \pm 0.67\%$	$\mathbf{77.05 \pm 0.65\%}$	$58.84 \pm 0.66\%$
		w/ LRP* [25]	$69.97 \pm 0.69\%$	$46.57 \pm 0.64\%$	$70.90 \pm 0.68\%$	$59.49 \pm 0.64\%$
		w/ ATA [27]	$66.22 \pm 0.50\%$	$49.14 \pm 0.40\%$	$75.48 \pm 0.40\%$	$54.26 \pm 0.40\%$
		w/ AFA [28]	$68.25 \pm 0.50\%$	$49.28 \pm 0.50\%$	$76.21 \pm 0.40\%$	$55.67 \pm 0.40\%$
		w/ DAC (Ours)	$\mathbf{75.10 \pm 0.59\%}$	$\mathbf{51.30 \pm 0.63\%}$	$74.55 \pm 0.66\%$	$\mathbf{62.47 \pm 0.63\%}$

TABLE 2. Results of our pilot study on where to insert the DAC module. The best results are highlighted in bold. With the exception of ProtoNet + L2 + 5-way 5-shot, all the results suggest that inserting the DAC module after the *L*3 layer is the optimal choice.

		L1	L2	L3	L4
5-way 1-Shot	MatchingNet [1]	$42.64 \pm 0.58\%$	$42.99 \pm 0.62\%$	$\mathbf{44.53 \pm 0.60\%}$	$43.84 \pm 0.58\%$
	ProtoNet [2]	$42.08 \pm 0.58\%$	$40.97 \pm 0.59\%$	$\mathbf{42.67 \pm 0.59\%}$	$35.48 \pm 0.54\%$
	RelationNet [18]	$44.42 \pm 0.64\%$	$45.82 \pm 0.59\%$	$\mathbf{49.10 \pm 0.63\%}$	$44.99 \pm 0.63\%$
	GNNNet [3]	$48.51 \pm 0.72\%$	$48.65 \pm 0.70\%$	$\mathbf{53.06 \pm 0.75\%}$	$50.96 \pm 0.78\%$
5-way 5-Shot	MatchingNet [1]	$56.49 \pm 0.58\%$	$58.07 \pm 0.58\%$	$\mathbf{62.38 \pm 0.56\%}$	$49.39 \pm 0.52\%$
	ProtoNet [2]	$60.20 \pm 0.54\%$	$\mathbf{61.68 \pm 0.56\%}$	$60.33 \pm 0.58\%$	$51.74 \pm 0.52\%$
	RelationNet [18]	$62.58 \pm 0.58\%$	$61.44 \pm 0.58\%$	$\mathbf{65.96 \pm 0.55\%}$	$62.63 \pm 0.56\%$
	GNNNet [3]	$67.01 \pm 0.68\%$	$68.33 \pm 0.70\%$	$\mathbf{75.10 \pm 0.59\%}$	$72.75 \pm 0.65\%$

CUB-200, Cars196, Places365, and Plantae datasets. The remaining three datasets and mini-ImageNet were used as seen domains for the meta-training stage. In addition to pure baseline methods, we combined LFT and LRP into each baseline method to demonstrate their performance boost. We conducted experiments in 5-way 1-shot and 5-way 5-shot settings, and calculated the average classification accuracy of 1,000 episodes in the test stage as the performance metric. The quantitative results of our experiments are presented in Table 1.

In this table, there are 32 controlled experiments, and our model outperformed the other models in 26 of them. In most groups, our model is the winner, and it surpasses the second-best results by at least $1-2\%$. In some specific groups, such

TABLE 3. We take MatchingNet and GNNNet as baseline methods to demonstrate the importance of the feature disentanglement loss $\mathcal{L}$dis. In this table, "Baseline" refers to the original few-shot classification method. "Ours (w/o $\mathcal{L}$dis)" indicates the proposed DAC module is inserted but trained without the disentanglement loss term $\mathcal{L}_{\text{dis}}$. The results conclude that training the model with $\mathcal{L}_{\text{dis}}$ can always help the generalization ability in the unseen domain.

			CUB-200	Cars196	Places365	Plantae
5-way 1-Shot	MatchingNet [1]	Baseline	$37.90 \pm 0.55\%$	$28.96 \pm 0.45\%$	$49.01 \pm 0.65\%$	$33.21 \pm 0.51\%$
		Ours (w/o $\mathcal{L}_{\text{dis}}$)	$43.65 \pm 0.61\%$	$30.91 \pm 0.50\%$	$46.40 \pm 0.64\%$	$36.62 \pm 0.59\%$
		Ours	$\mathbf{44.53 \pm 0.60\%}$	$\mathbf{33.03 \pm 0.51\%}$	$\mathbf{49.95 \pm 0.65\%}$	$\mathbf{43.21 \pm 0.63\%}$
	GNNNet [3]	Baseline	$49.46 \pm 0.73\%$	$32.95 \pm 0.56\%$	$51.39 \pm 0.80\%$	$37.15 \pm 0.60\%$
		Ours (w/o $\mathcal{L}_{\text{dis}}$)	$49.90 \pm 0.70\%$	$32.27 \pm 0.53\%$	$51.72 \pm 0.76\%$	$39.68 \pm 0.63\%$
		Ours	$\mathbf{53.06 \pm 0.75\%}$	$\mathbf{35.34 \pm 0.56\%}$	$\mathbf{52.96 \pm 0.82\%}$	$\mathbf{43.51 \pm 0.69\%}$
5-way 5-Shot	MatchingNet [1]	Baseline	$51.92 \pm 0.80\%$	$39.87 \pm 0.51\%$	$61.82 \pm 0.57\%$	$47.29 \pm 0.51\%$
		Ours (w/o $\mathcal{L}_{\text{dis}}$)	$59.24 \pm 0.58\%$	$42.39 \pm 0.53\%$	$61.53 \pm 0.57\%$	$51.54 \pm 0.56\%$
		Ours	$\mathbf{62.38 \pm 0.56\%}$	$\mathbf{46.38 \pm 0.59\%}$	$\mathbf{65.67 \pm 0.55\%}$	$\mathbf{58.66 \pm 0.57\%}$
	GNNNet [3]	Baseline	$69.26 \pm 0.68\%$	$48.91 \pm 0.67\%$	$72.59 \pm 0.67\%$	$58.36 \pm 0.68\%$
		Ours (w/o $\mathcal{L}_{\text{dis}}$)	$73.65 \pm 0.67\%$	$45.65 \pm 0.61\%$	$73.15 \pm 0.64\%$	$59.17 \pm 0.65\%$
		Ours	$\mathbf{75.10 \pm 0.59\%}$	$\mathbf{51.30 \pm 0.63\%}$	$\mathbf{74.55 \pm 0.66\%}$	$\mathbf{62.47 \pm 0.63\%}$

as MatchingNet + Plantae + 5-way 1-shot and MatchingNet + Plantae + 5-way 5-shot, our model leads by a remarkable 10% and 11.37%, respectively.

C. ABLATION STUDY

Firstly, we validate the importance of our proposed feature disentanglement loss, denoted as $\mathcal{L}_{\text{dis}}$. To accomplish this, we select MatchingNet and GNNNet as our baseline methods and present the results of corresponding experiments in Table 3, where the models are trained on the CUB-200 dataset (1-shot task). It is noteworthy that the performance consistently declines when we exclude $\mathcal{L}_{\text{dis}}$ from our model.

Secondly, we argue that the placement of the DAC module has a significant impact on the performance of the model. If it's inserted too shallow, it may not effectively identify common information from complex texture information across different domains. Conversely, if it's inserted too deep, only semantic information specific to individual domains may be preserved. To balance the amount of texture information and semantic features, we conducted experiments to find the optimal position for inserting the DAC module.

Since the first two convolutional layers in ResNet-10 are used to transfer the channel numbers of input images, we tested the performance of the DAC module after the 4th, 6th, 8th, and 10th layers, denoted as $L1$, $L2$, $L3$, and $L4$, respectively, in both 5-way 1-shot and 5-way 5-shot experiments. We placed the DAC module in MatchingNet, ProtoNet, RelationNet, and GNNNet models in each position.

Our results, shown in Table 2, indicate that inserting the DAC module after the $L3$ layer is the optimal position, except for the ProtoNet model with 5-way 5-shot setting, where the optimal position is $L2$.

V. CONCLUSION

This paper argues that poorly estimated domain-specific features can lead to inadequate generalization on the unseen domain. We propose a disentanglement-and-calibration module (DAC) to address this issue. By exploring optimal placement, we insert the DAC module into four classical few-shot classification models and train them using our proposed strategy. We find that methods based on calibrating only domain-specific components can help the network learn more generalized and discriminative information, enabling the network to exhibit certain robustness when confronted with unseen datasets during training. Our extensive experiments demonstrate that our approach consistently improves performance to a significant degree. In the future, we will apply our method to Simulation-to-Reality task to overcome the problem of domain gap between the simulator and the real world.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to Akiko Fukuda and Xiao Wang for their unwavering support and confidence in their work.

REFERENCES

[1] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2016, pp. 1–9.

[2] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 30, 2017, pp. 4077–4087.

[3] V. Garcia and J. Bruna, "Few-shot learning with graph neural networks," 2017, *arXiv:1711.04043*.

[4] S. Hegselmann, A. Buendia, H. Lang, M. Agrawal, X. Jiang, and D. Sontag, "TabLLM: Few-shot classification of tabular data with large language models," in *Proc. Int. Conf. Artif. Intell. Statist. (PMLR)*, 2023, pp. 5549–5581.

[5] P. Chikontwe, S. Kim, and S. H. Park, "CAD: Co-adapting discriminative features for improved few-shot classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14534–14543.

[6] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. F. Wang, and J.-B. Huang, "A closer look at few-shot classification," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–17.

[7] H.-Y. Tseng, H.-Y. Lee, J.-B. Huang, and M.-H. Yang, "Cross-domain few-shot classification via learned feature-wise transformation," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–18.

[8] A. Gonzalez-Garcia, J. Van De Weijer, and Y. Bengio, "Image-to-image translation for cross-domain disentanglement," 2018, *arXiv:1805.09730*.

[9] Y. Fu, Y. Fu, and Y.-G. Jiang, "Meta-FDMixup: Cross-domain few-shot learning guided by labeled target data," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 5326–5334.
[10] X. Peng, Z. Huang, X. Sun, and K. Saenko, "Domain agnostic learning with disentangled representations," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 5102–5112.
[11] N. Dvornik, C. Schmid, and J. Mairal, "Selecting relevant features from a multi-domain representation for few-shot classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2020, pp. 769–786.
[12] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1126–1135.
[13] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017, pp. 1–8.
[14] A. A. Rusu, D. Rao, J. Sygnowski, O. Vinyals, R. Pascanu, S. Osindero, and R. Hadsell, "Meta-learning with latent embedding optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–17.
[15] C. Finn, K. Xu, and S. Levine, "Probabilistic model-agnostic meta-learning," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 31, 2018, pp. 9516–9527.
[16] A. Nichol and J. Schulman, "Reptile: A scalable metalearning algorithm," 2018, *arXiv: 1803.02999*.
[17] L. Bertinetto, J. F. Henriques, P. H. S. Torr, and A. Vedaldi, "Meta-learning with differentiable closed-form solvers," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–15.
[18] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1199–1208.
[19] B. Hariharan and R. Girshick, "Low-shot visual recognition by shrinking and hallucinating features," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3037–3046.
[20] Y. Wang, R. Girshick, M. Hebert, and B. Hariharan, "Low-shot learning from imaginary data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7278–7286.
[21] Y. Guo, N. C. Codella, L. Karlinsky, J. V. Codella, J. R. Smith, K. Saenko, T. Rosing, and R. Feris, "A broader study of cross-domain few-shot learning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2020, pp. 124–141.
[22] P. Mangla, M. Singh, A. Sinha, N. Kumari, V. N. Balasubramanian, and B. Krishnamurthy, "Charting the right manifold: Manifold mixup for few-shot learning," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 2207–2216.
[23] L. Liu, W. Hamilton, G. Long, J. Jiang, and H. Larochelle, "A universal representation transformer layer for few-shot image classification," 2020, *arXiv:2006.11702*.
[24] J. Cai, B. Cai, and S. M. Shen, "SB-MTL: Score-based meta transfer-learning for cross-domain few-shot learning," 2020, *arXiv:2012.01784*.
[25] J. Sun, S. Lapuschkin, W. Samek, Y. Zhao, N. Cheung, and A. Binder, "Explanation-guided training for cross-domain few-shot classification," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 7609–7616.
[26] C. P. Phoo and B. Hariharan, "Self-training for few-shot transfer across extreme task differences," 2020, *arXiv:2010.07734*.
[27] H. Wang and Z.-H. Deng, "Cross-domain few-shot classification via adversarial task augmentation," 2021, *arXiv:2104.14385*.
[28] Y. Hu and A. J. Ma, "Adversarial feature augmentation for cross-domain few-shot classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 20–37.
[29] G. Blanchard, G. Lee, and C. Scott, "Generalizing from several related classification tasks to a new unlabeled sample," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 24, 2011, pp. 2178–2186.
[30] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," 2021, *arXiv:2103.02503*.
[31] H. Li, S. J. Pan, S. Wang, and A. C. Kot, "Domain generalization with adversarial feature learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 5400–5409.
[32] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto, "Unified deep supervised domain adaptation and generalization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5716–5726.
[33] K. Muandet, D. Balduzzi, and B. Schölkopf, "Domain generalization via invariant feature representation," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2013, pp. 10–18.
[34] Y. Li, X. Tian, M. Gong, Y. Liu, T. Liu, K. Zhang, and D. Tao, "Deep domain generalization via conditional invariant adversarial networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 647–663.
[35] R. Shao, X. Lan, J. Li, and P. C. Yuen, "Multi-adversarial discriminative deep domain generalization for face presentation attack detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10023–10031.
[36] Z. Wang, M. Loog, and J. van Gemert, "Respecting domain relations: Hypothesis invariance for domain generalization," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 9756–9763.
[37] L. Niu, W. Li, and D. Xu, "Multi-view domain generalization for visual recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4193–4201.
[38] R. Volpi, H. Namkoong, O. Sener, J. C. Duchi, V. Murino, and S. Savarese, "Generalizing to unseen domains via adversarial data augmentation," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 31, 2018, pp. 5334–5344.
[39] S. Shankar, V. Piratla, S. Chakrabarti, S. Chaudhuri, P. Jyothi, and S. Sarawagi, "Generalizing across domains via cross-gradient training," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–12.
[40] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, "Learning to generalize: Meta-learning for domain generalization," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 3490–3497.
[41] Y. Balaji, S. Sankaranarayanan, and R. Chellappa, "MetaReg: Towards domain generalization using meta-regularization," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 31, 2018, pp. 1006–1016.
[42] Y. Li, Y. Yang, W. Zhou, and T. M. Hospedales, "Feature-critic networks for heterogeneous domain generalization," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 3915–3924.
[43] Q. Liu, Q. Dou, and P.-A. Heng, "Shape-aware meta-learning for generalizing prostate MRI segmentation to unseen domains," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, 2020, pp. 475–485.
[44] Y. Du, X. Zhen, L. Shao, and C. G. M. Snoek, "MetaNorm: Learning to normalize few-shot batches across domains," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1–23.
[45] Y. Zhao, Z. Zhong, F. Yang, Z. Luo, Y. Lin, S. Li, and N. Sebe, "Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 6273–6282.
[46] S. Choi, T. Kim, M. Jeong, H. Park, and C. Kim, "Meta batch-instance normalization for generalizable person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3424–3434.
[47] W. Liu, Z. Liu, Z. Yu, B. Dai, R. Lin, Y. Wang, J. M. Rehg, and L. Song, "Decoupled networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 2771–2779.
[48] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "β-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017, pp. 1–22.
[49] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, vol. 29, Dec. 2016, pp. 2180–2188.
[50] H. Hwang, G.-H. Kim, S. Hong, and K.-E. Kim, "Variational interaction information maximization for cross-domain disentanglement," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 22479–22491.
[51] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.
[52] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-UCSD birds 200," Tech. Rep., 2010.
[53] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2013, pp. 554–561.
[54] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, Jun. 2018.
[55] G. Van Horn, O. M. Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie, "The iNaturalist species classification and detection dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 8769–8778.

HAO ZHENG received the B.S. degree from the Nanjing University of Science and Technology, in 2019, and the M.S. degree from The Chinese University of Hong Kong, in 2020. He is currently pursuing the Ph.D. degree, under the supervision of Prof. Kanezaki. His current research interests include transfer learning, few-shot classification, and domain adaptation problems.

QIANG ZHANG is currently a Visiting Researcher with the Brain Inspired Computing Laboratory, The Hong Kong University of Science and Technology, Guangzhou. His current research interests include robotic reinforcement learning, artificial intelligence, computer vision, and machine learning.

ASAKO KANEZAKI (Member, IEEE) received the B.S., M.S. and Ph.D. degrees in information science and technology from The University of Tokyo, in 2008, 2010, and 2013, respectively. In 2010, she was a Visiting Researcher with the Intelligent Autonomous Systems Group, Technische Universität München. From 2013 to 2016, she was an Assistant Professor with The University of Tokyo. She was with the National Institute of Advanced Industrial Science and Technology (AIST), from 2016 to 2020. Since 2020, she has been an Associate Professor with the Tokyo Institute of Technology. Her current research interests include object detection, 3D shape recognition, and robot applications, such as semantic mapping and visual navigation.