Disentangled Feature Representation for Few-Shot Image Classification

Hao Cheng, Yufei Wang, Haoliang Li, Alex C. Kot, Life Fellow, IEEE, and Bihan Wen, Member, IEEE

Abstract—Learning the generalizable feature representation is critical to few-shot image classification. While recent works exploited task-specific feature embedding using meta-tasks for few-shot learning, they are limited in many challenging tasks as being distracted by the excursive features such as the background, domain, and style of the image samples. In this work, we propose a novel disentangled feature representation (DFR) framework, dubbed DFR, for few-shot learning applications. DFR can adaptively decouple the discriminative features that are modeled by the classification branch, from the class-irrelevant component of the variation branch. In general, most of the popular deep few-shot learning methods can be plugged in as the classification branch, thus DFR can boost their performance on various few-shot tasks. Furthermore, we propose a novel FS-DomainNet dataset based on DomainNet, for benchmarking the few-shot domain generalization (DG) tasks. We conducted extensive experiments to evaluate the proposed DFR on general, fine-grained, and cross-domain few-shot classification, as well as few-shot DG, using the corresponding four benchmarks, i.e., mini-ImageNet, tiered-ImageNet, Caltech-UCSD Birds 200-2011 (CUB), and the proposed FS-DomainNet. Thanks to the effective feature disentangling, the DFR-based few-shot classifiers achieved state-of-the-art results on all datasets.

Index Terms—Deep learning, feature disentangling, few-shot classification, image classification.

I. INTRODUCTION

WILE deep neural networks achieved superior results on image classification via supervised learning from large-scale datasets, it is challenging to classify a query sample using only few labeled data, which is known as few-shot classification [1]. How to learn the discriminative feature representation that can be generalized from the training set to new classes in testing is critical to few-shot tasks. Popular few-shot methods applied meta-learning [2] by episodic training from a large number of simulated meta-tasks, to obtain a task-specific feature embedding associated with a distance metric (e.g., cosine or Euclidean distance) for classification.

Fig. 1 shows two such examples in fine-grained and multidomain classification tasks, respectively, which are challenging for few-shot learning: 1) only the subtle traits are critical to characterize and differentiate the objects of fine-grained classes and 2) the style and domain information dominates the image’s visual presence, but they are in fact the excursive and class-irrelevant features. As the subtle traits vary in different simulated meta-tasks, they can hardly be preserved by the learned embedding. On the contrary, the excursive features usually distract the feature embedding [3], [4], leading to degraded few-shot classification results. To rectify such limitations, most recent few-shot methods attempted to suppress excusive features or to propose proper metrics, e.g., learning compositional representation method (LCR) [3], deep Earth movers’ distance (DeepEMD) [4], few-shot embedding adaptation with transformer (FEAT) [5], and CNL [6]. However, none of the existing methods explicitly extract the class-discriminative representation from the excusive image features.

In this article, we present a novel approach to incorporate deep disentangling for few-shot image classification. Such an approach can selectively extract the subtle traits for each task while maintaining the model generalization. First, we propose a novel disentangled feature representation (DFR) framework which can be applied to most few-shot learning methods. DFR contains two branches: the classification branch extracts the discriminative features of the image sample, while the variation branch encodes the class-irrelevant information that complements the image representation. A relation module is applied in the variation branch to measure the feature dissimilarity.
similarity of each sample pair. A hybrid loss is applied for training DFR, including a reconstruction loss to ensure image information preservation, as well as the translation, discriminative, and cross-entropy losses for class-discriminative feature disentangling. At the inference stage, only the disentangled features from the classification branch are used for class prediction. Second, we integrate the proposed DFR framework into representative baselines for few-shot classification, e.g., the popular ProtoNet [7] and the state-of-the-art DeepEMD [4], FEAT [5], and feature map reconstruction network (FRN) [8], to carefully investigate the behavior of DFR with feature visualization and analysis. Extensive experiments are conducted on a set of few-shot tasks, i.e., general image classification, fine-grained image classification, cross-domain image classification, and domain generalization (DG) over four benchmarks to demonstrate the effectiveness of our DFR framework.

Our main contributions are summarized as follows.

1) We propose a novel DFR framework, which can be easily applied to most of the few-shot learning methods to extract class-discriminative features from excessive information.

2) We propose two evaluation settings to comprehensively investigate the few-shot DG task, using our proposed novel benchmark named FS-DomainNet, which is reorganized from DomainNet [9].

3) We evaluate the DFR framework over four few-shot benchmarks, i.e., mini-ImageNet, tiered-ImageNet, Caltech-UCSD Birds 200-2011 (CUB), and the proposed FS-DomainNet dataset. Results show that incorporating DFR into existing few-shot algorithms, including the baseline and state-of-the-art methods, can generate consistent improvement for multiple few-shot classification tasks under both five-way one-shot and five-way five-shot settings.

II. RELATED WORK

A. Few-Shot Learning

According to the meta-learning framework [2], there are mainly three types of few-shot learning methods.

1) Gradient-Based Methods: Gradient-based methods utilize a good model initialization [10], [11] or optimization strategy [12], [13], [14], [15], [16], [17], [18], [19], [20] to quickly adapt to novel tasks. One of the pioneering works is model-agnostic meta-learning (MAML) [10], which learns a set of network weights as initialization to adapt to new tasks quickly. To further optimize MAML, meta-learning with task-adaptive loss (MeTAL) [19] adopted meta-learners to learn a task-adaptive loss function during the inner-loop optimization of MAML to achieve better generalization. To alleviate the problem of catastrophic forgetting for few-shot continuous learning, Lee et al. [20] proposed attention-dependent mechanisms (AIMs), which combined the idea of adaptive learning with the decoupling of deep network feature extraction and high-order concept learning.

2) Data Augmentation-Based Methods: Data augmentation-based methods focus on generating [21], [22] or gathering augmented data [23], [24], [25], [26], [27], [28] to alleviate the data insufficiency. For example, Schwartz et al. [26] adopted an auto-encoder to extract transferable deformations between pairs of training samples, which can then be applied to generate novel samples from unseen classes to achieve data augmentations. Additionally, some recent works learned to estimate the data distribution to achieve data augmentation. For instance, Zhang et al. [24] proposed a variational Bayesian framework to approximate class-discriminative distributions via variational inference, suppressing the class-irrelevant information for novel input. Yang et al. [28] calibrated the distribution of few sample classes in the feature space by transferring statistics from the classes with sufficient base examples to avoid over-fitting problems caused by the biased distribution. Tang et al. [29] proposed a generation operator BlockMix to calibrate the category prototype to enhance metric learning, which generalizes well to the novel classes.

3) Metric Learning-Based Methods: In this work, we focus on the third type, namely the metric learning-based methods [4], [7], [17], [30], [31], [32], [33], [34], [35], [36], [37], i.e., to learn the discriminative feature embeddings for distinguishing different image classes. For example, ProtoNet [7] considered the class-mean representation as the prototype of each class and applied the Euclidean distance metric for classification. Memory matching network (MMNet) [38] utilized a memory module to refine the embedding of the support set extracted from matching network [2]. Deep nearest neighbor neural network (DN4) [30] replaced the image-level features with local descriptors and leveraged the k-nearest neighbor (KNN) to select k most relevant patches in each class for prediction on a given query sample. LCR [3] applied the subspace-based embedding for each class, and DeepEMD [4] adopted the earth mover’s distance as the metric function to compare the similarity between two feature maps in a structured way. FEAT [5] defined four kinds of set-to-set transformation, including self-attention transformer [39] to learn a task-specific feature embedding for few-shot learning. Based on prior knowledge, concept learners with meta-learning method (COMET) [40] mapped some high-level visual concepts into a semi-structured metric space and then learned an ensemble classifier by combining the outputs of independent concept learners. Tang et al. [41] also use a semi-structured feature space based on independent prior knowledge concepts to do pose normalization for fine-grained tasks. SetFeat [37] embedded shallow self-attention mechanisms inside existing encoder architectures to combine the output feature map of each convolution to build a discriminative representation of images.

In addition to the above-mentioned methods, there are also some other methods to solve the few-shot problem from different perspectives, e.g., knowledge transfer [42], [43] and batch normalization (BN) [44], [45]. Knowledge transfer network (KTN) [42] proposed a GCN-based semantic-visual mapping network to jointly integrate visual and semantic information for knowledge transfer. Knowledge-guided graph routing (KGGGR) [43] adopted two graph propagation mechanisms to exploit prior knowledge to guide adaptive information propagation among different classes for multilabel few-shot classification. MetaNorm [44] utilized a meta-learning approach.
to predict domain-specific BN statistics to address domain shift. Yazdanpanah et al. [45] proposed feature normalization to adjust channel distribution to replace BN during source domain training to improve the network robustness against distribution shifts.

Our work does not intend to propose new metrics but focuses on extracting the class-discriminative features from the variations distancing the metric learning, thus improving few-shot classification performance.

### B. Disentangled Feature Representations

DFR aims to learn an interpretable representation for image variants, which has been widely studied in tasks such as face generation [46], style translation [47], [48], image restoration [49], video prediction [50], and image classification [51], [52], [53]. InfoGAN [46] applied an unsupervised method to learn interpretable and disentangled representations by maximizing mutual information. DRIT [47] embedded images into a content space and a domain-specific attribute space and applied a cycle consistency loss for style translation. FDR [49] applied channel-wise feature disentanglement to reduce the interference between hybrid distortions for hybrid-distorted image restoration. Li et al. [52] proposed a disentangled variational auto-encoder (VAE) to excavate category-distilling information from visual and semantic features for generalized zero-shot learning. Zhang et al. [24] proposed a variational Bayesian framework to approximate class-discriminative distributions via variational inference, suppressing the class-irrelevant information for novel input. D3DP [51] adopted a feature disentangling scheme for few-shot detection and visual question-answering tasks in 3-D scenes, by dividing high-dimensional data (e.g., RGB-D) into individual objects and other attributes.

It is noteworthy that the very recent variational feature disentangling (VFD) [53] also adopted a feature disentanglement scheme for few-shot fine-grained classification. However, our DFR significantly differs from VFD in the following aspects. First, our DFR is a general few-shot framework that can be applied to most few-shot learning methods by introducing additional loss terms to achieve feature disentangling. In contrast, the VFD is a data augmentation scheme that generates features by modeling intraclass variance based on VAE. Second, DFR achieves feature disentanglement by inverting the gradients of relation modules in the variation branch, unlike the VAE-based VFD, which requires any distribution priors. However, our DFR can be applied to various few-shot scenarios and tasks, including general, fine-grained, cross-domain, and DG classifications, while VFD only focuses on the fine-grained classification task. Thus, DFR can effectively combine existing backbones to enhance the extracted features and be widely used in most few-shot methods.

### III. PROPOSED METHOD

In this section, we start with an introduction to few-shot learning. Then, the proposed DFR framework is described and explained in detail, followed by the loss function of our model and why our DFR works well.

#### A. Problem Definition

Given a training image set with the base classes $C_{\text{train}}$, few-shot image classification task aims to predict the novel classes $C_{\text{test}}$ from the testing set, i.e., $C_{\text{train}} \cap C_{\text{test}} = \emptyset$. Thus, the trained classifier from $C_{\text{train}}$ needs to be generalized to $C_{\text{test}}$ in the testing stage with only a few labeled samples. In this article, we follow the meta-learning strategy (i.e., the $N$-way $K$-shot setting) [2] to simulate meta-tasks in the training set that are similar to the few-shot setting at the testing stage, i.e., each meta-task $T_{\text{FS}}$ contains a support set $S$, and a query set $Q$. The support set $S$ contains $N$ classes with $K$ labeled samples (both $N$ and $K$ are very small), and a query set $Q$ with unlabeled query samples from $N$ classes is used to evaluate the performance.

#### B. Disentangled Feature Representation

Fig. 2 is an overview of the proposed DFR framework. With a few-shot task $T_{\text{FS}}$ of support set $S$ and query set $Q$,
the objective is to extract discriminative features of the image sample $X_i$, for classification from the exclusive information of each feature $x_i$. The proposed DFR consists of two branches with two encoders, i.e., $E_{cls}$ and $E_{var}$ for the feature extractor and variation branches, respectively, and one decoder $D$, as well as a discriminator with a gradient reverse layer and a relation module RM.

1) Classification Branch: In principle, any classic metric-based methods for few-shot learning can be applied in this branch based on feature extractor $E_{cls}$ to extract the class-discriminative features $x_i = E_{cls}(X_i)$ of each $X_i$ for classification. In this work, the commonly used ResNet-12 backbone is adopted as $E_{cls}$, and the classifier $f(\cdot)$ varies for different few-shot learning baselines being used (e.g., ProtoNet [7], DeepEMD [4], and FEAT [5] are applied in this work, with the corresponding models denoted as +DFR). Therefore, the query sample $X^Q_i$ can be classified based on the support samples $X^S$ as follows:

$$\tilde{y}_i = f(E_{cls}(X^Q_i); \{E_{cls}(X^S), y^S\})$$

(1)

2) Variation Branch: The role of the variation branch is to separate the class-irrelevant information from the feature of image samples $X$, which consists of an encoder $E_{var}$ followed by a discriminator. The discriminator is formed by a gradient reversal layer (GRL) and a relation module $r_v$ to measure the variation feature similarity between any two samples. Specifically, the GRL acts as an identity transform in the forward pass, and multiplies the gradient from the subsequent level by a constant $-\lambda$ during back-propagation. In training, first, we extract the class-irrelevant feature $v_i$ with the encoder $E_{var}$ and compute the class prototype $c_i$ based on feature vectors $z_i$ for each sample $X_i$ in the variation and classification branch, respectively. Then, the relation module $r_v$ takes these two terms as input and outputs the probability distribution of sample $X_i$ over $N$ classes based on cosine or Euclidean distance. The probability of predicting $v_i$ as the kth class is

$$s^j_i = p(y = j | v_i) = \frac{\exp\left(-d(v_i, c^j)\right)}{\sum_{k=1}^{N} \exp(-d((v_i, c^k)))}$$

(2)

where $d(\cdot)$ is the distance metric function and $c_k = (1/K) \sum_{j=1}^{K} z_k$ is the prototype (i.e., the mean feature vector) of the kth class.

3) Decoder Module: To preserve the image information and achieve feature disentanglement, a decoder $D$ combines the classification and variation branches for feature reconstruction and translation.

Specifically, the decoder $D$ takes a pair of feature vectors from two branches as the input and reconstructs or translates the original features based on the source of class-relevant information for each sample $X_i$. During training, $D$ learns to reconstruct the features of each sample $X_i$ from the kth class based on the sample-wise feature $z_i$ of $x_i$ as follows:

$$\hat{x}_i = D(v_i, z_i).$$

(3)

In addition to using $z_i$ for reconstruction at the sample level, we also achieve reconstruction and translation at the class level with the prototype $c_k$ of the kth class as follows:

$$\hat{x}_i^k = D(v_i, c_k)$$

(4)

where we set $k = y_i$ for reconstruction and $k \neq y_i$ for translation.

C. Loss Function

The objective function consists of the discriminative loss $L_{dis}$, cross-entropy loss $L_{cls}$, reconstruction loss $L_{rec}$, and translation loss $L_{tran}$.

1) Discriminative Loss: To remove the class-discriminative information in the variation branch, we adopt the relation module $r_v$ for class prediction based on class prototypes computed in the classification branch with the cross-entropy loss as follows:

$$L_{dis} = -\sum_{i=1}^{M} y_i \log s_i$$

(5)

where $M$ denotes the number of training samples, $s_i$ represents the probability distribution of sample $X_i$ over $N$ classes calculated by (2), and $y_i$ denotes the true class label of each sample $X_i$.

Note that directly optimizing $L_{dis}$ will narrow the distance between the class-irrelevant feature $v_i$ and the prototype $c_i$ of the ground-truth class, which is the opposite of our goal. Therefore, we apply GRL to reverse the gradient during back-propagation to achieve feature disentangling, i.e., removing the class-discriminative information captured by the variation feature.

2) Cross-Entropy Loss: To preserve task-specific class-discriminative features for few-shot classification, we minimize the cross-entropy loss $L_{cls}$ for the classification branch for query samples of all classes in each meta-task as follows:

$$L_{cls} = -\sum_{i=1}^{Q} y_i \log P(\hat{y}_i = y_i | T_{FS})$$

(6)

where $Q$ is the number of query samples in a meta-task $T_{FS}$, and $y_i$ and $\hat{y}_i$ represent the true and predicted class label of each query sample $X_i$, respectively.

3) Reconstruction and Translation Loss: To ensure that the disentangled features from classification and variation branches can jointly restore the entire information of the input image, an $\ell_1$-norm penalty for image reconstruction at sample and class levels is applied as follows:

$$L_{rec} = \frac{1}{M} \sum_{i=1}^{M} \|x_i - \hat{x}_i\|_1 + \frac{1}{M} \sum_{i=1}^{M} \|x_i - \hat{x}_i^y\|_1$$

(7)

where $M$ denotes the number of samples in a meta-task $T_{FS}$, $\hat{x}_i$ and $\hat{x}_i^y$ are the reconstructed feature maps of $x_i$ based on the feature of the $i$th sample itself and mean feature $c_i$ of class $y_i$ using (3) and (4), respectively.

Moreover, the cross-entropy loss is also adapted to measure perceptual differences between the output translated feature map $\hat{x}_i^y$ and the prototype of the $j$th class $c_j$ for class translation to achieve feature disentanglement as follows:

$$L_{tran} = -\sum_{i=1}^{Q} y_i \log \frac{\exp(-d(\phi(\hat{x}_i^y), c_j))}{\sum_{k=1}^{N} \exp(-d((\hat{x}_i^y), c_k))}$$

(8)

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
where \( \phi(\cdot) \) is the global average pooling operation and \( \hat{x}_i^j \) is the translated feature map of \( x_i \) based on the prototype of the \( j \)th class.

The total loss for training DFR can be formulated as follows:

\[
L_{\text{total}} = \lambda_1 \cdot L_{\text{dis}} + \lambda_2 \cdot L_{\text{rec}} + \lambda_3 \cdot L_{\text{tran}} + L_{\text{cls}} \tag{9}
\]

where \( \lambda_1, \lambda_2, \) and \( \lambda_3 \) denote the weight parameters of \( L_{\text{dis}}, L_{\text{rec}}, \) and \( L_{\text{tran}} \) relative to \( L_{\text{cls}} \), respectively.

D. Why It Works

DFR framework aims to extract only class-relevant information for classification. Different from other attempts toward more adaptive embedding using attention mechanism [5], [56], [61], our classification and variation branches play as the adversaries by minimizing \( L_{\text{cls}} \) and \( L_{\text{dis}} \) simultaneously. In practice, the classification and variation features of an image are always complimentary. Thus, the image reconstruction quality is enforced after fusion by minimizing \( L_{\text{rec}} \). It is essential to preserve the image representation in DFR for few-shot classification. As the class-discriminative features can be task-varying and thus hard to be generalized, any information loss throughout the interstage flow may potentially limit the model performance. Such design is in contrast to the classic feature embedding for few-shot learning, in which image features are always projected onto the lower-dimensional manifolds [59]. Classification feature \( z_i \) has a much lower dimension compared to the original feature \( x_i \), as the class-irrelevant information (e.g., image style, and background) are typically excessive. To this end, a more restrictive classification feature will significantly reduce the model bias, thus enhancing its generalizability in few-shot tasks.

1) Visualization Analysis: We visualize the feature distributions without and with DFR using \( t \)-SNE [70] to verify our intuition. Fig. 3(a) shows that the learned features extracted from the ResNet-12 backbone are less discriminative without using the DFR framework. While when applying the DFR framework, the classification branch clusters in Fig. 3(b) are more separable from each other, and the output features of the variation branch in Fig. 3(c) contain more class-irrelevant information that meets our expectations.

To qualitatively evaluate the proposed DFR framework, we also provide the class activation mapping (CAM) visualization results [71] of two DFR branches on novel classes of the mini-ImageNet dataset as shown in Fig. 4. As shown in Fig. 4, the variation branch attends more to the background information for the target objects \( \text{ant} \) and \( \text{bowl} \). In contrast, the classification branch captures objects with rich semantic information, which can generalize better to novel tasks. The results demonstrate that the DFR framework can deactivate fewer class-discriminative features via the gradient reverse layer in the variation branch, e.g., the image regions of neck and legs for “golden retriever” and “dalmation,” respectively.

IV. Experiment

In this section, we first describe the implementation details of our networks. Then, we conduct extensive experiments on two few-shot benchmarks, i.e., Mini-ImageNet, and Tiered-ImageNet on general few-shot classification tasks to evaluate the performance of our proposed DFR framework. After that, we introduce a novel FS-DomainNet dataset with the proposed two evaluation settings for benchmarking the few-shot DG task, dubbed FS-DG. Moreover, we evaluate the performance of DFR on the CUB-200-2011 benchmark on fine-grained few-shot classification task. We also conduct cross-domain experiments to validate the generalizability of the proposed DFR framework. Finally, we perform ablation studies to analyze our DFR framework in detail.

A. Implementation Details

For a fair comparison with the existing few-shot learning methods, we adopt ConvNet-4 [2], [5], [7], ResNet-12 [14], and ResNet-18 [57] as our backbone \( E_{\text{cls}} \) for feature extraction. We set the numbers of channels as \([64, 64, 64], [64, 160, 320, 640], [64, 128, 256, 512]\) for ConvNet-4, ResNet-12, and ResNet-18, respectively, which are the same as the competing methods. The encoder \( E_{\text{var}} \) consists of three convolutional blocks. The decoder \( D \) consists of a fully connected block followed by three convolutional blocks, similar to \( E_{\text{var}} \). The I/O channel numbers of the variation encoder and decoder are all set to the output channel number of the backbone. The level ratio \( \lambda \) of the GRL is set to 0.5 for the mini-ImageNet dataset and 0.1 for other datasets.

Data augmentation, including resizing, random cropping, color jitter, and random flipping [5], are applied for all methods in training. Our models are all trained using SGD optimizer, with the weight decay as \( 5e^{-4} \), and the momentum as 0.9.

We conduct experiments under both five-way one-shot and five-way five-shot settings with 15 query images each class, i.e., \( 5 \times 1(5) + 5 \times 15 \) samples for one-shot and five-shot tasks, respectively. For the \( N \)-way \( K \)-shot evaluation, we report the mean accuracy of randomly sampled 600 tasks as well as the 95% confidence intervals on the testing set with 15 query samples for each class per task. To verify the effectiveness of our proposed DFR framework, we combined DFR with four few-shot algorithms: a commonly used baseline ProtoNet [7], three state-of-the-art methods DeepEMD [4],

1We adopt the code of the Grad-CAM [72] method implemented in PyTorch on https://github.com/jacobgil/pytorch-grad-cam

2The code of the proposed DFR framework and FS-DomainNet dataset will be available on https://github.com/chengcv/DFRFS
Fig. 4. CAM visualization on five test classes (corresponding to each row) of the mini-ImageNet dataset. Here, the columns of “original,” “classification,” and “variation” represent the original image, CAM visualization of the classification branch, and CAM visualization of the variation branch of the DFR framework, respectively. CAM is produced by using Grad-CAM [72].

| Method   | mini-ImageNet | tiered-ImageNet | CUB | FE-DomainNet |
|----------|---------------|----------------|-----|--------------|
| lr       | step          | lr             | step| lr           | step          | lr       | step          | lr       | step          |
| ProtoNet [7] | 0.001 | 10             | 0.001 | 10/30       | 0.001 | 10       | 0.001 | 10       |
| FEAT [5]   | 0.0001 | 60/70          | 0.0002 | 40          | 0.0002 | 40       | 0.0002 | 40       |
| DeepEMD [6] | 0.0005 | 10             | 0.0005 | 10          | 0.0005 | 10       | 0.0005 | 10       |

Note that we only adopt the FCN version of DeepEMD for comparison over all datasets. Specifically, the initial learning rate (lr) and step size of ProtoNet, DeepEMD, and FEAT methods are shown in Table I.

Note that for DFR+FRN, the initial learning rates of mini-ImageNet and tiered-ImageNet are 0.0001 and 0.0005, with step sizes 60 and 30, respectively. Moreover, for the CUB dataset, we follow the training strategy of FRN and adopt a three-stage end-to-end training with an initial learning rate of 0.1 and 600 epochs for each stage. Furthermore, the weighting parameters $\lambda_1$, $\lambda_2$, and $\lambda_3$ are set to 0.1, 1.0, and 1.0 for all datasets.

B. General Few-Shot Classification

We first conduct experiments on two general few-shot benchmarks: mini-ImageNet and tiered-ImageNet.

TABLE II

| Method                        | mini-ImageNet | tiered-ImageNet | ConvNet-4 Backbone |
|-------------------------------|---------------|-----------------|--------------------|
| lr                            | step          | lr              | step              |
| Baseline [54]                 | 48.24 ± 0.75  | 66.41 ± 0.65    | 51.67 ± 1.81      | 70.30 ± 1.75      |
| MAML [10]                     | 48.70          | 63.11           |                    |                    |
| RelationNet [55]              | 50.44 ± 0.82   | 65.32 ± 0.70    | 54.48 ± 0.93      | 71.32 ± 0.78      |
| MetaOptNet [14]               | 52.87          | 68.78           | 54.71             | 71.76             |
| ProtoNet [7]                  | 52.45 ± 0.20   | 71.14 ± 0.36    | 41.81 ± 0.29      | 60.13 ± 0.19      |
| ProtoNet + DFR                | 53.79 ± 0.83   | 71.88 ± 0.65    | 47.46 ± 0.87      | 64.67 ± 0.78      |
| bDeepEMD [4]                  | 43.89 ± 0.27   | 57.78 ± 0.71    | 41.37 ± 0.28      | 54.53 ± 0.79      |
| DeepEMD + DFR                 | 44.72 ± 0.77   | 58.71 ± 0.72    | 42.78 ± 0.78      | 55.74 ± 0.77      |
| FRN [8]                       | 49.63 ± 0.79   | 65.76 ± 0.65    | 54.47 ± 0.93      | 73.08 ± 0.74      |
| FRN + DFR                     | 50.02 ± 0.82   | 66.04 ± 0.65    | 55.04 ± 0.89      | 73.50 ± 0.71      |
| FL/FEAT [5]                   | 54.81 ± 0.20   | 71.53 ± 0.16    | 46.58 ± 0.21      | 61.10 ± 0.19      |
| FEAT + DFR                    | 55.44 ± 0.83   | 71.98 ± 0.64    | 48.27 ± 0.85      | 63.77 ± 0.79      |

1) Datasets: The mini-ImageNet dataset [2] is a subset of the ILSVRC-12 challenge [73] proposed for few-shot classification. It contains 100 diverse classes with 600 images of size $84 \times 84 \times 3$ in each category. Following the class split setting [12] used in previous works, all 100 classes are divided into 64, 16, and 20 classes for training, validation, and testing, respectively.

Similar to mini-ImageNet, tiered-ImageNet [74] is also a subset of ILSVRC-12, which contains more challenging classes that are organized in a hierarchical structure, i.e., 608 classes from 34 top categories. We follow the setups proposed in [74] and split 608 categories into 351, 97, and 160 for training, validation, and testing, respectively.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
2) Results: The classification results with different backbones are shown in Tables II, III, and IV.

We observe that the proposed DFR can consistently improve the performance of all combined methods under different backbones. The results also illustrate that the existing few-shot methods can achieve the best performance on almost all datasets (competitive with PSM under the five-shot setting over mini-ImageNet with the ResNet-12 backbone) after combining our proposed DFR, proving the effectiveness of the proposed DFR. Moreover, by adopting the DFR framework, the five-way one-shot accuracies by ProtoNet are increased by 3.51% and 3.32% on mini-ImageNet and tiered-ImageNet, respectively, which are even comparable to more sophisticated methods. For the other three few-shot methods, DeepEMD, FRN, and FEAT, which are the current state-of-the-art few-shot methods, their classification results can still be further boosted by 1%–2% on average after applying the DFR framework. It is worth noting that the performance gain of each method under the five-way five-shot setting is smaller than that under the one-shot setting. One plausible reason is that more supervised information introduced by support samples (i.e., from one-shot to five-shot) forces the few-shot methods to capture most class-relevant information, which helps reduce the intraclass variation, which is consistent with our idea.

It is worth noting that the performance gain of each few-shot method varies with different backbones. In general, DFR can improve classification performance under a more complex backbone. A reasonable explanation is that a deep backbone, e.g., ResNet-12, can extract more image information, and DFR can effectively help remove some irrelevant information for classification, thereby helping to improve performance. In contrast, a simple backbone like ConvNet-4 can extract limited feature information, restricting the capability of DFR.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
TABLE VII  
FINE-GRAINED FEW-SHOT CLASSIFICATION ACCURACY (%) AVERAGED ON CUB WITH THE RESNET-12 BACKBONE USING BOUNDING-BOX CROPPED IMAGES AS INPUT. “◊” DENOTES THE RESULTS REPORTED IN [8]. “♦” DENOTES THE RESULTS REPORTED IN [75].

| Method | CUB |
|--------|-----|
|        | 5-way 1-shot | 5-way 5-shot |
| RelationNet [55] | 66.20 ± 0.99 | 82.30 ± 0.58 |
| MAML [10] | 67.28 ± 0.18 | 83.47 ± 0.59 |
| MatchNet◊ [2] | 71.87 ± 0.85 | 85.08 ± 0.57 |
| COMET [40] | 72.20 ± 0.90 | 87.60 ± 0.50 |
| DUAL ATT-NET [78] | 72.89 ± 0.50 | 86.60 ± 0.31 |
| P-Transfer [62] | 73.88 ± 0.92 | 87.81 ± 0.48 |
| Curvature Generation [66] | 74.66 ± 0.21 | 88.37 ± 0.12 |
| BML [68] | 76.21 ± 0.63 | 90.45 ± 0.36 |
| PSM [67] | 77.43 ± 0.61 | 86.43 ± 0.43 |
| Tang et al. [77] | 78.73 ± 0.84 | 89.77 ± 0.47 |
| VDF [53] | 79.12 ± 0.83 | 91.48 ± 0.39 |
| ProtoNet [7] | 72.25 ± 0.21 | 87.47 ± 0.13 |
| ProtoNet+DFR | 75.74 ± 0.85 | 88.51 ± 0.52 |
| DeepEMD [4] | 74.88 ± 0.30 | 88.52 ± 0.52 |
| DeepEMD+DFR | 75.73 ± 0.82 | 89.30 ± 0.94 |
| FEAT [5] | 75.68 ± 0.20 | 87.91 ± 0.13 |
| FEAT+DFR | 78.07 ± 0.79 | 89.74 ± 0.51 |
| FRN [8] | 82.90 ± 0.19 | 92.61 ± 0.10 |
| FRN+DFR | 85.27 ± 0.19 | 93.16 ± 0.10 |

C. Fine-Grained Few-Shot Classification

We further evaluate DFR on a fine-grained benchmark, i.e., CUB [80], which was initially proposed for fine-grained image classification, which contains 200 different birds with 11,788 images. Following the split in [54] and [81], 200 classes are divided into 100, 50, and 50 for training, validation, and testing, respectively. Note that there are two different ways to preprocess the data: adopting the provided bounding box to crop the image [4], [5], [8] or directly using the raw image as input [8], [54]. To fully verify the effectiveness of the proposed DFR framework, we train and evaluate few-shot methods based on two settings. For CUB with cropped images as input, we conduct experiments on ConvNet-4, ResNet-12, and ResNet-18 backbones for evaluation. For CUB with raw images as input, we only adopt the ResNet-12 backbone following FRN [8].

1) Results: Tables V–VII report the fine-grained few-shot classification results over CUB under five-way one-shot and five-way five-shot settings with cropped images as input. Table VIII reports the results on the CUB dataset with raw images as input. Compared to the general few-shot benchmarks that contain significant differences between the categories, fine-grained classification only includes minor intra-class differences. The domain information in the fine-grained dataset may contribute to the category, making it a challenging task. It is obvious that the proposed DFR can also significantly and consistently boost all few-shot methods with different backbones for both cropped and raw images on the CUB dataset.

Moreover, comparing the results under two settings (Table VII with cropped images and Table VIII with raw images) with the same backbone ResNet-12, we can find that (a) few-shot methods perform worse under the second setting, since the original images contain more class-irrelevant information; (b) DFR can bring more performance improvements for raw images. One possible reason is that DFR can help to capture more discriminative regions and remove excursive features.

Above all, results demonstrate that DFR can effectively remove the excursive features, and thus highlight the subtle traits which are critical to fine-grained few-shot classification.

D. Cross-Domain Few-Shot Classification

To further validate the generalizability of the proposed DFR framework, we conduct the experiments on the cross-domain scenarios: mini-ImageNet → CUB, which is more challenging due to the large domain gap between the two datasets. We combine our proposed DFR framework with existing few-shot learning methods and compare them with baseline methods in Table IX. As we can see, the proposed DFR framework can consistently boost the performance of all combined methods. This result shows that DFR can provide more transferable information across datasets by removing the domain-relevant features and maintaining class-irrelevant features. Moreover, we can observe that the original ProtoNet performs better than FEAT under the five-shot setting, possibly because the self-attention transformer module in FEAT is susceptible to highly relevant features in fine-grained tasks. However, after applying our proposed DFR framework, the performance gap shrinks from 4.01% to 1.15%. This demonstrates that the proposed DFR can help extract more class-relevant features in the classification branch, with more significant differences across classes than features extracted from the variation branch.

E. FS-DomainNet Dataset for Benchmarking Few-Shot DG

1) Few-Shot DG: Compared to the general classification task that training and testing are based on the data with the same distribution, the DG task is more challenging due to...
the domain shift between source and target data. DG methods aim to learn a domain-agnostic model from multiple sources that can classify data from any target domain to overcome the domain shift problem. However, general DG setting assumes that training and testing data share the same class set, which may be hard to generalize to the novel class set in the few-shot scenario. Similarly, general few-shot learning does not consider the influence caused by the domain gap, making few-shot models hardly generalize to unseen domains in real scenarios.

In this work, we consider a more challenging Few-Shot DG (FS-DG) problem, i.e., both domain and class gaps exist between the training (source) and testing (target) sets. As shown in Fig. 6 and Table X, we divide the dataset into training and testing sets at the domain and class levels and test the performance with two evaluation settings. More details about the proposed dataset and experimental settings will be given in Sections IV-E2 and IV-E3, respectively.

2) FS-DomainNet Benchmark: We propose the FS-DomainNet dataset for benchmarking few-shot DG by reorganizing the original DomainNet [9], initially proposed for the multisource domain adaptation task. We reorganize it for few-shot learning and select all categories (i.e., 527 156 images of 299 classes) that include at least the number of samples (i.e., 20) required by the five-shot setting on each domain, i.e., five labeled support images and 15 query images for each class. Then, we split 299 categories into 191, 47, and 61 for training, validation, and testing, respectively, while maintaining the consistency of class split on each domain. Fig. 5 shows image examples of six selected categories from six distinct domains on the FS-DomainNet dataset. Unlike existing few-shot benchmarks, FS-DomainNet includes objects collected from multiple domains considering both the domain and class gaps. Moreover, the sample size varies significantly between categories to enable more challenging FS-DG task settings. Additionally, the FS-DomainNet dataset can also be utilized for general few-shot classification tasks.

Note that the previous work [44] also reorganizes the DomainNet for few-shot learning. Different from its split [44], which only contains 200 classes with 1000 images for each class, FS-DomainNet captures a much larger subset of DomainNet [9], i.e., 569 010 images with 345 different categories.

3) Experimental Setups: Following the classic DG setting, we choose five out of six domains from FS-DomainNet as the source domains and the remaining one as the target domain in our experiments. We report the average FS-DG accuracies over the splits with each of the six domains as the target domain.

a) Meta-Training Setting: For five-way one-shot tasks, we randomly select one support sample only from one random source domain for each class. For five-way five-shot tasks, we select one labeled sample of each source domain for each class. For five-way one-shot tasks, we randomly select one support sample only from one random source domain for each class. For five-way five-shot tasks, we select one labeled sample of each source domain for each class. For five-way five-shot tasks, we select the same number of support samples of each domain. For query samples of each class in the source domains under both five-way one-shot and five-shot settings, we select the same number of query samples (i.e., 3) from each domain, i.e., $\|Q\| = 3 \times 5 = 15$.

b) Evaluation Settings: According to the domain of support set $S$ during the inference time, we propose two

| Method | 5-way 1-shot | 5-way 5-shot |
|--------|--------------|--------------|
| MAML$^\dagger$ [10] | - | 51.34 ± 0.72 |
| MatchingNet$^\dagger$ [2] | - | 53.07 ± 0.74 |
| RelationNet$^\dagger$ [55] | - | 57.71 ± 0.73 |
| Baseline++$^\dagger$ [54] | - | 62.04 ± 0.76 |
| Baseline$^\dagger$ [54] | - | 65.57 ± 0.70 |
| ProtoNet [7] | 45.22 ± 0.21 | 66.29 ± 0.14 |
| ProtoNet+DFR | 46.72 ± 0.84 | 67.26 ± 0.72 |
| FEAT [5] | 45.33 ± 0.20 | 62.28 ± 0.51 |
| FEAT+DFR | 47.51 ± 0.80 | 66.11 ± 0.72 |
| DeepEMD [4] | 51.72 ± 0.29 | 77.01 ± 0.67 |
| DeepEMD+DFR | 52.69 ± 0.84 | 78.44 ± 0.68 |
| FRN [8] | 56.35 ± 0.21 | 77.09 ± 0.15 |
| FRN+DFR | 57.74 ± 0.21 | 79.97 ± 0.16 |

Fig. 5. Image samples of six selected categories (columns) from six distinct domains (rows) on the FS-DomainNet dataset.
**Fig. 6.** Illustration of training and two evaluation settings on the FS-DomainNet dataset, best viewed in color. Here, we take a five-way one-shot task with one query sample of each class as an example. We follow the basic few-shot setting by separating the training and testing set with non-overlap classes (i.e., borders of different colors—blue and brown). The main difference between the two evaluation settings is whether the support and query set come from the same target domain (i.e., with the same background color). Best viewed in color.

**TABLE XI**

| Classification Accuracy (%) Averaged on FS-DomainNet Under Five-Way Tasks with Evaluation Setting A (Support Set From the Target Domain) |
|-----------------------------------------------|
| Method                                      | Sketch(0) 1-shot | Sketch(0) 5-shot | Quickdraw(1) 1-shot | Quickdraw(1) 5-shot | Real(2) 1-shot | Real(2) 5-shot | Painting(3) 1-shot | Painting(3) 5-shot | Clipart(4) 1-shot | Clipart(4) 5-shot | Infographt(5) 1-shot | Infographt(5) 5-shot | Average |
|-----------------------------------------------|-----------------|-----------------|---------------------|---------------------|----------------|----------------|---------------------|---------------------|-----------------|-----------------|---------------------|---------------------|---------|
| MatchNet [2]                                 | 36.35±0.17      | 40.00±0.15      | 55.23±0.22          | 68.44±0.19          | 57.45±0.22      | 70.17±0.17      | 44.90±0.21          | 51.80±0.18          | 48.04±0.20      | 55.59±0.18      | 29.28±0.14         | 37.50±0.16         | 45.23    | 54.92    |
| ProtoNet [7]                                 | 38.46±0.18      | 40.74±0.68      | 59.14±0.22          | 63.54±0.71          | 59.68±0.22      | 71.78±0.71      | 47.91±0.21          | 51.63±0.73          | 52.89±0.21      | 63.66±0.71      | 29.70±0.15         | 37.33±0.71         | 47.96    | 56.95    |
| ProtoNet+DFR [82]                            | 39.64±0.75      | 41.22±0.72      | 60.31±0.93          | 70.38±0.64          | 62.48±0.88      | 82.18±0.56      | 51.24±0.93          | 71.38±0.66          | 55.33±0.89      | 75.82±0.61      | 31.18±0.63         | 44.68±0.69         | 30.06    | 69.05    |
| DeepE2MD [4]                                 | 38.41±0.66      | 47.78±0.71      | 59.71±0.89          | 70.28±0.64          | 71.02±0.86      | 86.44±0.50      | 55.41±0.87          | 73.22±1.04          | 60.31±0.85      | 79.80±0.55      | 32.31±0.67         | 47.01±0.73         | 52.86    | 70.59    |
| DeepE2MD+DFR [82]                            | 41.66±0.69      | 58.82±0.72      | 62.63±0.89          | 80.93±0.61          | 71.42±1.00      | 87.28±0.51      | 56.70±0.85          | 74.25±1.07          | 63.85±0.91      | 81.43±0.56      | 33.65±0.69         | 46.32±0.73         | 54.99    | 71.84    |
| FEAT [5]                                     | 39.71±0.18      | 59.00±0.69      | 59.86±0.22          | 77.41±0.66          | 66.60±0.22      | 82.53±0.55      | 54.02±0.22          | 71.75±0.69          | 58.17±0.21      | 77.38±0.61      | 32.60±0.17         | 44.85±0.74         | 51.83    | 68.82    |
| FEAT+DFR [82]                                | 43.82±0.78      | 60.93±0.65      | 60.53±0.89          | 78.60±0.65          | 67.59±0.88      | 83.12±0.53      | 59.42±0.88          | 72.85±1.07          | 60.48±1.05      | 87.31±0.58      | 33.06±0.67         | 46.13±1.05         | 54.15    | 69.91    |

FS-DG evaluation settings (i.e., FS-DG setting A and B) to evaluate the generalizability of the few-shot models by considering both domain and class gaps for few-shot methods, shown in Fig. 6 and Table X.

c) Setting A (Few-Shot DG Setting): As shown in the second row in Fig. 6, support samples are only sampled from the testing class set (i.e., in the brown border) from the target domain (i.e., in the yellow background).

d) Setting B (Few-Shot Cross-Domain Setting): As shown in the bottom row in Fig. 6, support samples are sampled from the testing class set (i.e., in the brown border) but from the source domains (i.e., in the blue background, same with training samples). Unlike the existing few-shot cross-domain settings in [82] and [83] that only follow the general few-shot setting on a new dataset, our proposed setting B is more challenging due to the domain gap between support and query set at test time. The challenge of this setting is how to transfer the learned knowledge of the source domain to the novel class set.

Both settings can evaluate the generalizability of the model, i.e., the ability to extract domain-invariant and class-discriminative features for classification. Furthermore, setting A is more challenging than setting B, as it forces the few-shot models to learn general knowledge from the domain and class levels. Recent works [5], [44] also attempted simple FS-DG tasks to evaluate their proposed few-shot models. However, only preliminary results are reported following the simple setting (i.e., Setting B in Table X) without a comprehensive investigation of the effect of the domain gap on novel classes (test class set). Note that the previous FS-DG setting proposed by meta-norm [44] contains one image of each meta-source domain for each class, i.e., $5 \times 5 = 25$ support samples for the five-way one-shot setting. In contrast, we only select one sample from one random domain for each class during the training time, i.e., $5 \times 1 = 5$, strictly following the general few-shot setting. For the query set, we fixed the total number of query samples to 75, while meta-norm tried a more extensive set of query examples as 125 to improve the performance. Moreover, we also consider two different domain-generalization evaluation settings to study the effect of domain information better.

4) Results: Tables XI and XII show the classification accuracy of six different target domains and the average result on the FS-DomainNet benchmark for two evaluation
settings, respectively. It is obvious that DFR can consistently improve classification accuracies for all FS baselines under both settings, thanks to its effective disentanglement of class-discriminative features.

It is worth noting that the baseline ProtoNet method without DFR has an accuracy of 56.95% and 54.77% under the five-shot setting. After including the proposed DFR framework, the performance increases by a large margin of 12.10% and 16.53%. A potential explanation is that the DFR framework can effectively disentangle class-discriminative features from overall features extracted from the backbone network. Hence, even the baseline ProtoNet method can obtain comparable results with the state-of-the-art few-shot methods. Furthermore, results show that our improvement over the other two SOTA methods, FEAT and DeepEMD, is more pronounced in the one-shot setting than in the five-shot setting. The reason is that we can only select one sample from one random domain for each class. In this case, we may see part of the source domain, making it more challenging to learn domain-specific information.

Meanwhile, another observation is that DeepEMD achieves a more significant improvement when combined with the proposed DFR framework than the FEAT method. It is partially due to the unique design of DeepEMD by adapting the channel-wise EMD metric based on the feature maps, which inextricably incorporates the similarity of domain information. In contrast, FEAT learns a self-attention block to extract task-specific features with extra parameters, which overlap to some extent with our ideas.

Furthermore, comparing the results of the two tables, we can observe that the second setting is indeed more challenging than the first, which is in line with our expectations. However, an interesting observation is that the proposed DFR framework can boost higher performance for this complex scenario. This result demonstrates that the DFR-based method can extract domain-invariant features of the support set from the source domains in the inference time, which is more discriminative for classification.

### F. Ablation Study

To investigate the effectiveness of the key components in our proposed DFR framework, we conduct extensive ablation studies by neglecting the loss function and compare the results by incorporating FEAT and ProtoNet as the baseline method on the mini-ImageNet dataset, respectively. The detailed results are shown in Tables XIII and XIV.

Results show that each loss term plays an essential part in our proposed DFR framework and improves the classification performance under both five-way one-shot and five-shot settings. Tables XIII and XIV validate the necessity of each loss term in our proposed DFR framework. The detailed value of each loss term represents the weighting parameters $\lambda_1$, $\lambda_2$, and $\lambda_3$, respectively.

### TABLE XII

| Method | Sketch(0) | Quickdraw(1) | Real(2) | Painting(3) | Clipart(4) | Infograph(5) | Average |
|--------|-----------|-------------|---------|-------------|-----------|-------------|---------|
|        | 1-shot    | 5-shot      | 1-shot  | 5-shot      | 1-shot    | 5-shot      | 1-shot  | 5-shot |
| MatchNet [2] | 32.94±0.16 | 37.72±0.15 | 40.76±0.20 | 52.56±0.20 | 39.83±0.19 | 42.91±0.21 | 56.93±0.19 | 44.16±0.22 | 56.64±0.18 | 43.08±0.21 | 37.41±0.16 | 40.61±0.49 |
| ProtoNet [7] | 35.65±0.17 | 42.93±0.68 | 47.32±0.20 | 52.12±0.74 | 51.14±0.21 | 54.34±0.75 | 52.45±0.21 | 58.47±0.75 | 53.04±0.22 | 60.70±0.78 | 52.55±0.22 | 60.04±0.72 | 48.70±0.57 |
| ProtoNet+DFR | 37.35±0.68 | 56.50±0.76 | 48.72±0.88 | 66.77±0.77 | 54.42±0.94 | 73.08±0.67 | 55.78±0.89 | 75.57±0.71 | 55.97±0.92 | 76.93±0.67 | 55.54±0.99 | 79.97±0.66 | 51.10±0.73 |
| DeepEMD [4] | 36.16±0.70 | 53.39±0.69 | 51.52±0.87 | 69.44±0.74 | 53.93±0.89 | 72.05±0.68 | 57.03±0.91 | 75.49±0.67 | 57.16±0.93 | 76.39±0.65 | 56.70±0.94 | 75.86±0.63 | 52.08±0.70 |
| DeepEMD+DFR | 37.93±0.72 | 56.06±0.67 | 53.54±0.87 | 70.96±0.69 | 58.49±0.92 | 76.10±0.67 | 60.47±0.93 | 77.97±0.66 | 61.27±0.91 | 79.04±0.63 | 60.64±0.96 | 78.66±0.62 | 55.39±0.73 |
| FEAT [5] | 36.75±0.17 | 54.25±0.73 | 50.05±0.21 | 67.62±0.76 | 55.61±0.21 | 73.36±0.68 | 57.28±0.22 | 76.08±0.68 | 57.69±0.22 | 77.37±0.64 | 57.37±0.22 | 76.43±0.68 | 52.46±0.70 |
| FEAT+DFR | 40.26±0.77 | 57.69±0.72 | 50.74±0.85 | 68.83±0.74 | 56.37±0.92 | 75.71±0.66 | 59.37±0.91 | 77.37±0.64 | 60.42±0.92 | 78.37±0.66 | 60.46±0.91 | 77.59±0.67 | 54.60±0.70 |

### TABLE XIII

**Ablation Study on the Mini-ImageNet Dataset of ProtoNet With the Proposed DFR Framework Under ResNet-12 Backbone. $\lambda$ Denotes the Gradient Reversal Factor in GRL. The Detailed Value of Each Loss Term Represents the Weighting Parameters $\lambda_1$, $\lambda_2$, and $\lambda_3$, Respectively.**

### TABLE XIV

**Ablation Study on the Mini-ImageNet Dataset of FEAT With the Proposed DFR Framework. $\lambda$ Denotes the Gradient Reversal Factor in GRL. The Detailed Value of Each Loss Term Represents the Weighting Parameters $\lambda_1$, $\lambda_2$, and $\lambda_3$, Respectively.**
In contrast, DFR with $L_{dis}$ and $L_{rec}$ can consistently improve the classification performance of combined methods, ProtoNet (3.35% for one-shot, and 0.63 for five-shot) and FEAT (0.73% for five-shot). On this basis, introducing an additional translation loss $L_{tran}$ can further ensure feature disentangling by constraining the transfer of discriminative information between images of different categories.

In Table XIII, another observation is that the improvement of DFR on one-shot (3.51%) is much larger than on five-shot (1.05%). Compared to the one-shot task with only one support sample for each class, in the five-shot setting, each class may have a larger intraclass variance. Therefore, the prototype obtained by simple feature averaging cannot accurately represent the class, which increases the difficulty of class-wise translation and reconstruction tasks.

To further evaluate the effect of hyper-parameter $\lambda$ for gradient reverse layer (GRL), we conduct experiments to evaluate the performance with different $\lambda \in [0, 1]$, shown in Fig. 7. We can observe that the model is not sensitive to the value of $\lambda$, and the proposed model can achieve the best performance when $\lambda = 0.5$.

V. CONCLUSION

We propose a novel and effective DFR framework for few-shot image classification. Unlike the feature embeddings, which may encode the excursive image information, such as background and domain, the proposed DFR aims to extract the task-specific class-discriminative features, which is essential in most few-shot learning pipelines. We have studied the importance of applying DFR in few-shot tasks by visualizing the CAM and t-SNE of the extracted features without DFR and disentangled features from the classification and variation branches. Furthermore, to tackle the challenges of the domain gap in few-shot learning, we propose a novel benchmarking dataset (FS-DomainNet) for the few-shot DG task. We thoroughly evaluate the generalizability of few-shot models with two proposed FS-DG evaluation settings on the FS-DomainNet dataset. Experimental results on four datasets, including four tasks (general image classification, fine-grained classification, cross-domain classification, and DG) under the few-shot settings, evaluate the effectiveness of the proposed DFR framework.

ACKNOWLEDGMENT

This work was done at Rapid-Rich Object Search (ROSE) Laboratory, Nanyang Technological University.

REFERENCES

[1] L. Fei-Fei, R. Fergus, and P. Perona, “One-shot learning of object categories,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 4, pp. 594–611, Apr. 2006.
[2] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, “Matching networks for one shot learning,” in Proc. 30th Int. Conf. Neural Inf. Process. Syst., 2016, pp. 3637–3645.
[3] P. Tokmakov, Y.-X. Wang, and M. Hebert, “Learning compositional representations for few-shot recognition,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 6372–6381.
[4] C. Zhang, Y. Cai, G. Lin, and C. Shen, “DeepEMD: Few-shot image classification with differentiable Earth mover’s distance and structured classifiers,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 12203–12213.
[5] H.-J. Ye, H. Hu, D.-C. Zhan, and F. Sha, “Few-shot learning via embedding adaptation with set-to-set functions,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 8808–8817.
[6] J. Zhao, Y. Yang, X. Lin, J. Yang, and L. He, “Looking wider for better adaptive representation in few-shot learning,” in Proc. AAAI Conf. Artif. Intell., vol. 35, no. 12, 2021, pp. 10981–10989.
[7] J. Snell, K. Swersky, and R. Zemel, “Prototypical networks for few-shot learning,” in Proc. 31st Int. Conf. Neural Inf. Process. Syst., 2017, pp. 4080–4090.
[8] D. Wertheimer, L. Tang, and B. Hariharan, “Few-shot classification with feature map reconstruction networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 8012–8021.
[9] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, “Moment matching for multi-source domain adaptation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 1406–1415.
[10] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in Proc. Int. Conf. Mach. Learn., 2017, pp. 1126–1135.
[11] A. Nichol, J. Achiam, and J. Schulman, “On first-order meta-learning algorithms,” 2018, arXiv:1803.02999.
[12] S. Ravi and H. Larochelle, “Optimization as a model for few-shot learning,” in Proc. Int. Conf. Learn. Represent., 2017. [Online]. Available: https://openreview.net/forum?id=ryKocll.
[13] A. A. Rusu et al., “Meta-learning with latent embedding optimization,” in Proc. Int. Conf. Learn. Represent., 2019. [Online]. Available: https://openreview.net/forum?id=BJghk8AeK7.
[14] K. Lee, S. Maji, A. Ravichandran, and S. Soatto, “Meta-learning with differentiable convex optimization,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 10657–10665.
[15] Y. Liu, B. Schiele, and Q. Sun, “An ensemble of epoch-wise empirical Bayes for few-shot learning,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Aug. 2020, pp. 404–421.
[16] H.-G. Jung and S.-W. Lee, “Few-shot learning with geometric constraints,” IEEE Trans. Neural Netw. Learn. Syst., vol. 31, no. 11, pp. 4660–4672, Nov. 2020.
[17] R.-Q. Wang, X.-Y. Zhang, and C.-L. Liu, “Meta-prototypical learning for domain-agnostic few-shot recognition,” IEEE Trans. Neural Netw. Learn. Syst., vol. 33, no. 11, pp. 6990–6996, Nov. 2022.
[18] P. Tian, W. Li, and Y. Gao, “Consistent meta-regularization for better meta-knowledge in few-shot learning,” IEEE Trans. Neural Netw. Learn. Syst., vol. 33, no. 12, pp. 7277–7288, Dec. 2022.
[19] S. Baik, J. Choi, H. Kim, D. Cho, J. Min, and K. M. Lee, “Meta-learning with task-adaptive loss function for few-shot learning,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 9465–9474.
[20] E. Lee, C.-H. Huang, and C.-Y. Lee, “Few-shot and continual learning with attentive independent mechanisms,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 9455–9464.
[21] S. Gidaris and N. Komodakis, “Generating classification weights with GNN denoising autoencoders for few-shot learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 21–30.
[22] K. Li, Y. Zhang, K. Li, and Y. Fu, “Adversarial feature hallucination networks for few-shot learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 13470–13479.
[68] Z. Zhou, X. Qiu, J. Xie, J. Wu, and C. Zhang, “Binocular mutual learning for improving few-shot classification,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 8402–8411.

[69] C. Liu et al., “Learning a few-shot embedding model with contrastive learning,” in Proc. AAAI Conf. Artif. Intell., vol. 35, no. 10, 2021, pp. 8635–8643.

[70] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE,” J. Mach. Learn. Res., vol. 9, no. 11, pp. 1–27, 2008.

[71] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Learning deep features for discriminative localization,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 2921–2929.

[72] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 618–626.

[73] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.

[74] M. Ren et al., “Meta-learning for semi-supervised few-shot classification,” in Proc. Int. Conf. Learn. Represent., 2018. [Online]. Available: https://openreview.net/forum?id=H1SzSzz-CZ

[75] X. Li, J. Wu, Z. Sun, Z. Ma, J. Cao, and Z-H. Xue, “BSNet: Bi-similarity network for few-shot fine-grained image classification,” IEEE Trans. Image Process., vol. 30, pp. 1318–1331, 2021.

[76] Y. Wu et al., “Object-aware long-short-range spatial alignment for few-shot fine-grained image classification,” in Proc. 29th ACM Int. Conf. Multimedia, Oct. 2021, pp. 107–115.

[77] H. Tang, C. Yuan, Z. Li, and J. Tang, “Learning attention-guided pyramidal features for few-shot fine-grained recognition,” Pattern Recognit., vol. 130, Oct. 2022, Art. no. 108792.

[78] S.-L. Xu, F. Zhang, X.-S. Wei, and J. Wang, “Dual attention networks for few-shot fine-grained recognition,” in Proc. AAAI Conf. Artif. Intell., 2022, vol. 36, no. 3, pp. 2911–2919.

[79] C. Doersch, A. Gupta, and A. Zisserman, “CrossTransformers: Spatially-aware few-shot transfer,” in Proc. Adv. Neural Inf. Process. Syst., vol. 33, 2020, pp. 21981–21993.

[80] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, “The Caltech-UCSD birds-200-2011 dataset,” Comput. Neural Syst., California Inst. Technol., Pasadena, CA, USA, Tech. Rep. 2010-001, 2011.

[81] N. Hilliard, L. Phillips, S. Howland, A. Yankov, C. D. Corley, and N. O. Hodas, “Few-shot learning with metric-agnostic conditional embeddings,” 2018, arXiv:1802.04376.

[82] 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., 2020. [Online]. Available: https://openreview.net/forum?id=SJ5Np4tPr

[83] Y. Guo et al., “A broader study of cross-domain few-shot learning,” in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2020, pp. 124–141.

**Hao Cheng** received the B.Eng. degree from the School of Software, Dalian University of Technology, Dalian, China, in 2016, and the M.S. degree from the School of Computer Science and Technology, Tianjin University, Tianjin, China, in 2019. He is currently pursuing the Ph.D. degree with Nanyang Technological University, Singapore. His research interests include few-shot learning, graph neural networks, and computer vision.

**Yufei Wang** received the B.Eng. degree from the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China, in 2020. He is currently pursuing the Ph.D. degree with Nanyang Technological University, Singapore. His research interests focus on image restoration and the generalization ability of deep neural network models. Dr. Wang was a recipient of the SDSC Dissertation Research Fellowship in 2022.

**Bihan Wen** (Member, IEEE) received the B.Eng. degree in electrical and electronic engineering from University of Electronic Science and Technology of China, Chengdu, China, in 2020. He is currently pursuing the Ph.D. degree with Nanyang Technological University, Singapore. His research interests include machine learning, computational imaging, computer vision, and video processing, and big data applications.

**Huoliang Li** received the Ph.D. degree from Nanyang Technological University (NTU), Singapore, in 2018. He is currently an Assistant Professor with the Department of Electrical Engineering, The City University of Hong Kong, Hong Kong. His research mainly focuses on AI security, multimedia forensics, and transfer learning. Dr. Li received the Wallenberg-NTU Presidential Post-Doctoral Fellowship in 2019, the Doctoral Innovation Award in 2019, the VCIP Best Paper Award in 2020, and the ACM SIGSOFT Distinguished Paper Award in 2022.

**Alex C. Kot** (Life Fellow, IEEE) has been with Nanyang Technological University, Singapore, since 1991. He was the Head of the Division of Information Engineering and the Vice Dean Research at the School of Electrical and Electronic Engineering. Subsequently, he served as an Associate Dean for College of Engineering for eight years. He is currently a Professor and the Director of Rapid-Rich Object Search (ROSE) Laboratory and NTU-PKU Joint Research Institute, Singapore. He has published extensively in the areas of signal processing, biometrics, image forensics and security, and computer vision and machine learning.

Dr. Kot was elected as the IEEE Distinguished Lecturer for the Signal Processing Society and the Circuits and Systems Society. He is a fellow of Academy of Engineering, Singapore. He received the Best Teacher of the Year Award and is a co-author for several Best Paper Awards including ICPR, the IEEE International Workshop on Information Forensics and Security, and the International Workshop on Digital Watermarking, Computer Vision and Pattern Recognition Precognition Workshop and VCIP. He served for the IEEE SP Society in various capacities such as the General Co-Chair for the 2004 IEEE International Conference on Image Processing and the Vice-President for the IEEE Signal Processing Society. He served as an Associate Editor for more than ten journals, mostly for IEEE TRANSACTIONS.