

# Free Lunch to Meet the Gap: Intermediate Domain Reconstruction for Cross-Domain Few-Shot Learning

Tong Zhang · Yifan Zhao\* · Liangyu Wang · Jia Li\*

Received: date / Accepted: date

**Abstract** Cross-Domain Few-Shot Learning (CDFSL) endeavors to transfer generalized knowledge from the source domain to target domains using only a minimal amount of training data, which faces a triplet of learning challenges in the meantime, *i.e.*, semantic disjoint, large domain discrepancy, and data scarcity. Different from predominant CDFSL works focused on generalized representations, we make novel attempts to construct **Intermediate Domain Proxies (IDP)** with source feature embeddings as the *codebook* and reconstruct the target domain feature with this learned *codebook*. We then conduct an empirical study to explore the intrinsic attributes from perspectives of *visual styles* and *semantic contents* in intermediate domain proxies. Reaping benefits from these attributes of intermediate domains, we develop a fast domain alignment method to use these proxies as learning guidance for target domain

feature transformation. With the collaborative learning of intermediate domain reconstruction and target feature transformation, our proposed model is able to surpass the state-of-the-art models by a margin on 8 cross-domain few-shot learning benchmarks. Our code and models will be publicly available.

**Keywords** Few-shot learning · Cross-domain · Intermediate domain reconstruction

## 1 Introduction

While current deep vision systems are undoubtedly successful at image classification tasks (He et al. 2016; Simonyan and Zisserman 2014), their exceptional performance heavily relies on the availability of large-scale labeled data. Although these large-scale datasets (Deng et al. 2009) are making progress in facilitating networks to reach higher performance, it is usually impractical to gather such vast amounts of data when dealing with a novel concept. This data scarcity problem has motivated the research on Few-Shot Learning (FSL) (Fei-Fei et al. 2006; Lake et al. 2015; Miller et al. 2000), which aims to model generalized memories from sufficient base training samples while tackling conceptually novel categories during the inference stage.

Despite its substantial improvements in many ideal tasks, few-shot learning approaches suffer from a default assumption: *the base pre-training categories for generalized knowledge and the few-shot novel categories should distribute in one same domain*. However, such strong assumptions are not feasible in most real-world systems, especially when exploring new concepts including remote-sensing satellite images (Helber et al. 2019) and medical images (Rajpurkar et al. 2017). To meet the gap in realistic usage, Cross-Domain Few-Shot

---

\*: Corresponding author

Tong Zhang

State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, China.

E-mail: tongzhang@buaa.edu.cn

Yifan Zhao

State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, China.

E-mail: zhaoyf@buaa.edu.cn

Liangyu Wang

State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, China.

E-mail: lyuwang@buaa.edu.cn

Jia Li

State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, China.

E-mail: jiali@buaa.edu.cn

Learning (CDFSL) is established when various levels of domain distribution shifts exist between pre-trained base classes and target novel classes. Pioneer studies (Chen et al. 2019; Guo et al. 2020) have demonstrated that predominant methods for FSL exhibit significant performance degradation when applied to the challenging problem of CDFSL. The huge gap between source and target domains impedes networks from extrapolating the generalized knowledge learned from source classes to target novel ones.

Recent approaches are devoted to learning a generalized representation to tackle this challenging problem, which generalizes the model from the source domain to the target domain. Tseng et al. (2020) propose to augment the features with randomness by using the feature transformation layer. Wang and Deng (2021a) augment multiple tasks in an adversarial manner. In comparison, Liang et al. (2021) introduce noisy distribution to enhance the network for learning robust image representations. Beyond these domain generalization methods, other research efforts (Li et al. 2022a; Shirekar et al. 2023) focus on mitigating the domain gap from source to target ones with the facilitation of few-shot samples, *e.g.*, Phoo and Hariharan (2020) propose to adapt the network to the target domain by performing self-supervised learning on the target domain. Nevertheless, this learning mechanism relies on the large additional amount of unlabeled data on the target domain, which is usually infeasible in certain practical scenarios.

One intuitive idea to address the CDFSL is to conduct few-shot learning with domain adaptation techniques, *e.g.*, MMD distance (Tzeng et al. 2014; Pan et al. 2010) or adversarial training (Ganin et al. 2016; Tzeng et al. 2017). However, different from the prevalent domain issues, the CDFSL tackles a triplet of learning challenges. 1) **Semantic disjoint**: the semantic label spaces of the source and target domains are mutually exclusive, which is commonly shared in typical domain adaptation problems. 2) **Domain discrepancy** between domains can be extremely large, such as the stark contrast between the visual characteristics of natural images (Deng et al. 2009) and X-ray images (Wang et al. 2017). 3) **Data scarcity**: the  $N$ -way  $K$ -shot FSL samples are substantially difficult to represent the target domain distributions. These triplet simultaneous challenges in CDFSL lead to a clear failure when using prevalent domain alignment techniques.

Keeping these challenges in mind, in this paper, we make an attempt to construct Intermediate Domain Proxies, forming a shared latent space between the source and target domains. Toward this end, we first form a prototypical vector pool learned from the embedding of source categories and select the representative vectors

to learn a mapping function from source to target features, *i.e.*, the dense source feature embedding serves as a codebook to reconstruct features of each target sample. We then transform the target samples with the learned mapping functions to construct an **intermediate domain**, which inherently follows two basic principles: 1) the intermediate domain shares the same *semantic content* as the target domains; 2) the *visual style* of intermediate domain achieves a good compromise between the source and target domains. Reaping benefits from these principles, the reconstructed features of the intermediate domain, namely proxies, exhibit fewer inherent domain gaps than their source counterparts while still retaining the visual cues of these sources. Instead of the naive alignment of source and target domains in prevailing works, we here use the intermediate domain proxies as the learning guidance to re-adjust both low and high-order parameters in feature normalization layers (*e.g.*, Batch Normalization (Ioffe and Szegedy 2015)). Hence under extreme data scarcity, the network features can be fast aligned to the target domains. In conclusion, these intermediate proxies conduct relaxed alignment constraints instead of the direct alignment of target and source domains. During the domain alignment phase, we propose a rapid feature transformation using these proxies without the rehearsal of source data, and during the inference phase, our method does not rely on the constructed intermediate domains. This lightweight implementation indicates the "free-lunch" design of the proposed approach. The main contributions of our work are three-fold:

1. We make attempts to construct Intermediate Domain Proxies (IDP) to solve the cross-domain few-shot learning problems and analyze the intrinsic attributes of these proxies from the perspective of visual styles and semantic content.
2. We develop a fast adaptation method to use intermediate domain proxies as learning guidance for target domain feature transformations.
3. We propose a unified framework for intermediate domain reconstruction and fast domain feature transformation in CDFSL. Experimental evidence indicates our proposed framework outperforms the state-of-the-art methods by a large margin on 8 public datasets.

The remainder of this paper is organized as follows: Section 2 describes the related works of this research and Section 3 presents an empirical study of the intermediate domain. Section 4 describes the proposed intermediate domain proxies reconstruction approach for the cross-domain few-shot learning problem. Qualita-

tive and quantitative experimental results are reported in Section 5. Section 6 finally concludes this paper.

## 2 Related Works

**Few-shot Learning** (FSL) aims to recognize novel concepts with very few numbers images, which are roughly categorized into two lines. The optimization-based approaches (Finn et al. 2017; Rusu et al. 2018; Vuorio et al. 2019; Li et al. 2017; Nichol et al. 2018) try to find a starting point for quickly optimizing the model. On the other hand, metric-based approaches (Oreshkin et al. 2018; Snell et al. 2017; Sung et al. 2018; Vinyals et al. 2016; Xu et al. 2022a; Chen et al. 2022) tend to find a task-independent embedding space that can be generalized to the target category by designing metric functions. To better capture detailed features of the image, recent metric-based approaches (Zhang et al. 2020a; Wertheimer et al. 2021; Ye et al. 2020) have focused on dense measuring of the feature map, rather than the global representation prototypes. Several methods (Rizve et al. 2021; Gidaris et al. 2019; Wei et al. 2022; Luo et al. 2021) set auxiliary tasks in the pre-training phase to enhance the model’s generalization ability. Dorsch et al. (2020) utilize the attention scheme to transfer the pretrained knowledge to few-shot learning. Nevertheless, these methods do not consider the significant differences between the source and target domains in the CDFSL setting.

**Domain Adaption** methods (Tzeng et al. 2017; Cui et al. 2020; Robey et al. 2021; Kang et al. 2018; Zhang et al. 2019) align the source and target domains to resolve domain shifts. These methods usually focus on regularizing the feature similarity of source and target domains, thus confusing the backbone networks to construct a unified representation for both domains. Another line of methods (Gong et al. 2012; Gopalan et al. 2013; Dai et al. 2021) propose reducing the domain alignment difficulty by designing intermediate domains to connect the source and target domains. Unlike these methods, the CDFSL task in this paper requires solving massive target domain classification tasks simultaneously with access to only a tiny number of samples from the supporting dataset. Simply mitigating the domain gaps would lead to severe overfitting on these few-shot samples.

**Cross-Domain Few-shot Learning** (CDFSL) methods (Guo et al. 2020; Wang and Deng 2021a; Phoo and Hariharan 2020; Liang et al. 2021) usually adopt the framework of transfer learning, *i.e.*, supervised learning using source domain data and then fine-tuning using target domain samples. This simple pipeline has proven


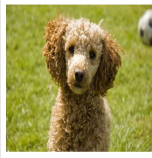


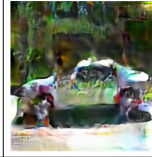



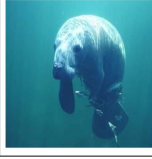
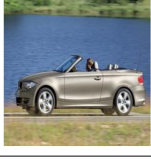

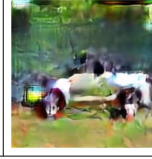
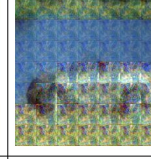

superior to many SOTA FSL methods in CDFSL settings (Chen et al. 2019; Guo et al. 2020). Apart from these works, Li et al. (2022b) focus on cross-domain domain generalization and fast adaptation to unseen domains with network tuning techniques, which stands at a different view from the conventional CDFSL learnings. Besides, Xu et al. (2022b) propose a contrastive learning scheme to distill the memorized knowledge from source domains. Representative methods (Phoo and Hariharan 2020; Li et al. 2022a; Shirekar et al. 2023) suggest enhancing cross-domain discrimination through either pseudo-labeling for model distillation or GNN-based message passing in the target domain. However, in these works (Phoo and Hariharan 2020; Li et al. 2022a; Shirekar et al. 2023), unlabeled target domain data are not always available in practical tasks, *e.g.*, X-ray images. Unlike these methods, we argue that in the fine-tuning phase, the network only has access to the target domain data is not the most effective use of the knowledge learned in the past due to catastrophic forgetting (McCloskey and Cohen 1989; Kemker et al. 2018). In contrast, in this paper, we propose to find a solution that only uses few-shot target domain data and without the rehearsal from the large-scale source domain data. With this in mind, we propose to build intermediate domain proxies instead of accessing additional source or target data.

**Discussions and Relations.** The concept of intermediate domain is proposed in domain adaptation and extended to many applications in previous studies, including Gong et al. (2012); Gopalan et al. (2013); Dai et al. (2021). However, these works require a huge demand for training samples of the target domain, which is unavailable for the few-shot scenarios. Thus we resort to the closed-form feature reconstruction methods (Bertinetto et al. 2018; Wertheimer et al. 2021) to build intermediate domains in the feature space using dense prototypical sources (Snell et al. 2017). By leveraging the advantage of the intermediate domain and feature reconstruction, the intermediate proxies are generated as a bridge when there are extremely few-shot available samples in target domains. Based on these proxies, beyond the conventional feature space alignment, our approach conducts a fast domain adaptation by normalization feature transformation techniques.

## 3 Intermediate Domain Reconstruction: An Empirical Study

### 3.1 Motivations and Setup

Imagine we are drawing an object, we almost always first draw the *sketch* of the object and then fill in the

|          |   |   | Reconstruction  | (A) img→r.img   | (A) feat.→r.feat.  | (B) img→r.img   | (B) feat.→r.feat.   |
|----------|---|---|---|---|--|---|---|
| Domain A |  |  |  |  |  |  |  |
| Class A  | <i>Birds, dogs and sloths</i>   |   |   |   |  |   |   |
| Domain B |  |  |  |  |  |  |  |
| Class B  | <i>Jellyfish, manatee and butterfly fish</i>                                      |   |   |   |  |   |   |
|          |   |   | Style   | <i>Jungle and meadow</i>  |  | <i>Underwater</i>   |   |

(a) sub-Domains

(b) Reconstructed Intermediate proxies

**Fig. 1** Illustration of intermediate proxy reconstruction. a) Two representative sub-domains for reconstruction bases. b) Reconstructed intermediate proxies using the sources in a).

color with various *pigments and inks*, finishing with wash, canvas, or watercolor styles. Analogous to this painting process, in CDFSL, we decouple the object representation into **semantic contents** and **visual styles**. Our motivation is to find an intermediate proxy domain  $\mathcal{P}$  that shares the same semantic features of samples in the target domain  $\mathcal{T}$  by using the base units in the source domain  $\mathcal{S}$ , forming the intermediate domain with a remix of both domains.

**Empirical Study.** Toward this motivation, we start from constructing the intermediate proxies with few visually similar images sampled from two “sub-Domains” in the *miniImageNet* (Vinyals et al. 2016), *i.e.*, i) Domain  $\mathcal{A}$ : *objects in jungle and meadow* including birds, dogs, and sloths; ii) Domain  $\mathcal{B}$ : *underwater objects* such as Jellyfish. We then take one exemplar image from Car dataset (Krause et al. 2013) as the target “domain”, which shows different contents that never appeared in the source domain, shown in Fig. 1.

**Construction of Intermediate Proxy** Suppose one domain as source domain  $\mathcal{S}$  and the other as target domain  $\mathcal{T}$ , we first extract feature bases  $\{\mathbf{C}_i\}_{i=1}^n \in \mathbb{R}^{n \times d}$  by network backbone  $f_\theta$ , where  $n$  denotes the number of features and  $d$  denotes their dimensions. To construct intermediate proxies, one intuitive method is to reconstruct the feature maps of each target sample using the elements in the source bases, serving as the codebook. Hence the reconstructed feature map could share both the source and target domain features, *i.e.*, reconstructed bases are similar to source domain  $\mathcal{S}$  and reconstructed goals are similar to target domain  $\mathcal{T}$ . In this paper, we name this process as the Intermediate Proxy Construction.

Let  $\mathbf{T} \in \mathbb{R}^{r \times d}$  denote the target domain embeddings required to be reconstructed. Following the Ridge Regression (Wertheimer et al. 2021; Hoerl and Kennard 1970; Bertinetto et al. 2018), and Sparse Coding mechanisms (Mairal et al. 2010), hence we need to find a mapping matrix  $\mathbf{W} \in \mathbb{R}^{r \times n}$  to minimize the reconstruction error  $\sum_{j=1}^r \|\mathbf{T}_j - \mathbf{W}\mathbf{C}\|_2^2$ . To retain this reconstruction as a convex process, we have the following Ridge regression form:

$$\widehat{\mathbf{W}} = \arg \min_{\mathbf{W}} \|\mathbf{T} - \mathbf{W}\mathbf{C}\|_2^2 + \lambda \|\mathbf{W}\|_2^2. \quad (1)$$

where  $\lambda$  is the hyper-parameter to balance the regularization of  $\ell_2$ -norm. With this regularization, we hereby use its closed-form solution as in (Bertinetto et al. 2018; Wertheimer et al. 2021), which also prevents the low-rank issues ( $n < r$ ) in solving Eq. (1), *i.e.*, making  $\widehat{\mathbf{W}}$  an invertible matrix:

$$\mathbf{P} = \widehat{\mathbf{W}}\mathbf{C} = \mathbf{T}\mathbf{C}^\top (\mathbf{C}\mathbf{C}^\top + \lambda\mathbf{I})^{-1}\mathbf{C}. \quad (2)$$

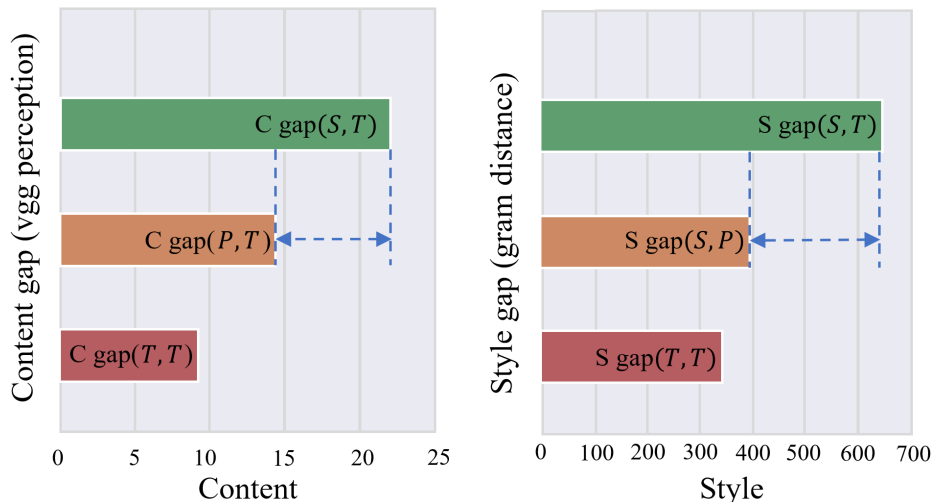
Here reconstructed intermediate proxies  $\mathbf{P} \in \mathbb{R}^{r \times d}$  have the same size as target domain features  $\mathbf{T}$ .

### 3.2 Analyzing Styles and Semantics of Intermediate Proxy

With the reconstructed intermediate proxies, here arise two inherent questions. **Q1**: *How does the choice of base  $\{\mathbf{C}_i\}_{i=1}^n$  impact the domain reconstruction?* and **Q2**: *What is the relationship of intermediate proxies with the source/target domains?*

To answer these two questions, here we conduct two lines of reconstruction with quantitative and qualitative analyses. *img*  $\rightarrow$  *r.img* denotes that we use





**Fig. 2** Quantitative evaluations of content and style differences among source domains  $\mathcal{S}$ , target domains  $\mathcal{T}$ , and intermediate domain proxies  $\mathcal{P}$ . The content distance is calculated by VGG perception and Style distance is calculated by Gram distance using samples in *mini-ImageNet* (Detailed in Appendix A).

source images to learn mapping weights  $\widehat{\mathbf{W}}$  on target images. And  $\text{feat} \rightarrow \mathbf{r}.\text{feat}$  denotes that we use the feature reconstruction process as in Eqs. (1) and (2). The feature extraction process with  $f_\theta$  uses the pretrained ResNet-10 backbone and other implementation details could refer to Section 5.1 and Appendix A. In Fig. 2, we calculate the content distance (VGG perceptual scores (Simonyan and Zisserman 2014)) and style variances (Gram distance (Gatys et al. 2015)) of among source  $\mathcal{S}$ , target  $\mathcal{T}$  and intermediate proxies  $\mathcal{P}$ . Here we draw three principles:

1. **The Intermediate Proxy preserves the semantic content of the target image.** In Fig. 1, the reconstructed images clearly shows the similar content *i.e.*, the cars with the target domain images. While the distance of content of  $(\mathcal{P}, \mathcal{T})$  is much more closer than  $(\mathcal{S}, \mathcal{T})$ .
2. **The Intermediate Proxy reflects the source domain style when using different bases.** Comparing the domain  $\mathcal{A}$  and  $\mathcal{B}$ , the reconstructed images show **green** background when using jungle images while exhibiting **blue** backgrounds using underwater images as bases.
3. **The Intermediate Proxy shows fewer domain shifts in content and style statistics than source domains.** Both the style and content shift in  $(\mathcal{P}, \mathcal{T})$  are much less than  $(\mathcal{S}, \mathcal{T})$  in Fig. 2, indicating aligning with intermediate domain proxies  $\mathcal{P} \rightarrow \mathcal{T}$  would be much easier than the direct alignment of source and target ones  $\mathcal{S} \rightarrow \mathcal{T}$ . Note that the distance to target domains could be closer if more base vectors are used for reconstruction.

### 3.3 The Role of Intermediate Proxy

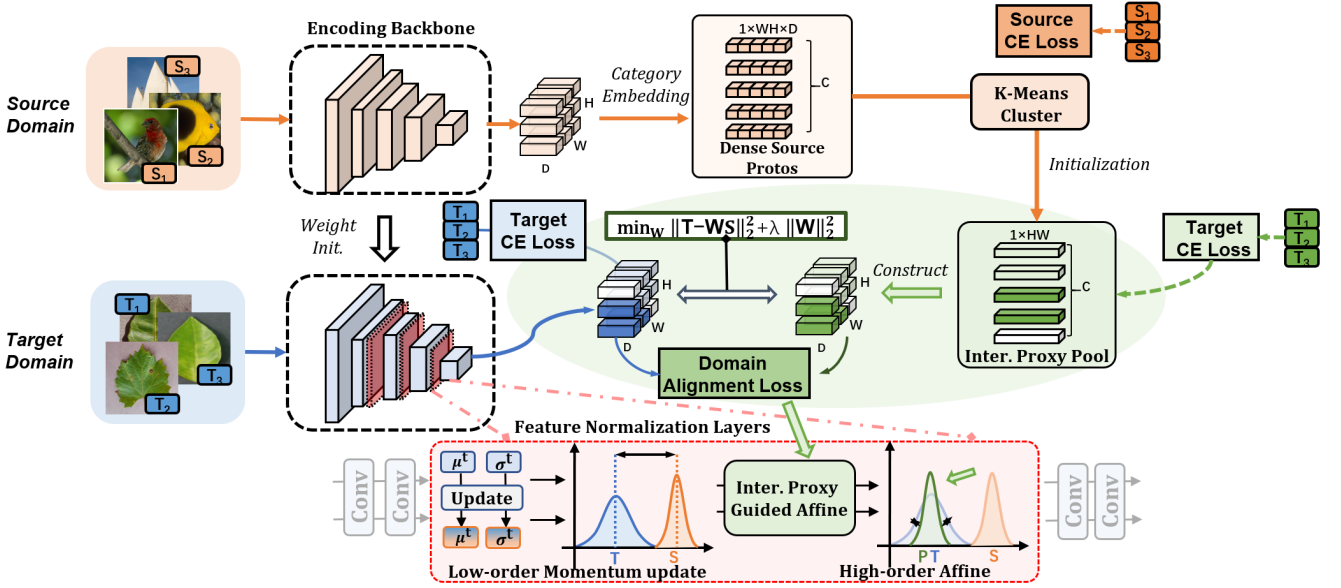
Besides the intuitive visualization and experimental statistics, here we provide detailed theoretical analyses of why the intermediate proxies help cross-domain learning.

**Definition 1** (*Inter-domain discrepancy distance*). *Inter-domain discrepancy distance between the source domain  $\mathcal{S}$  and target domain  $\mathcal{T}$  is measured by the Euclidean distance of their corresponding sample features:  $\text{disc}_{\mathcal{L}}(\mathcal{S}, \mathcal{T}) = \sum_{ij} \|\mathcal{S}_i - \mathcal{T}_j\|^2$ , with smaller discrepancy distances indicating greater semantic similarity between the domains.*

**Proposition 1** (*High semantic similarity*). *By controlling the ridge regression regular term  $\lambda$ , the semantic similarity between the intermediary domain proxy  $\mathcal{P}_\lambda$  and the target domain  $\mathcal{T}$  is larger than that between the source domain  $\mathcal{S}$  and the target domain  $\mathcal{T}$ . Their inter-domain discrepancy distance satisfies the following relationship:  $\exists \lambda, \text{s.t. } \text{disc}_{\mathcal{L}}(\mathcal{S}, \mathcal{T}) > \text{disc}_{\mathcal{L}}(\mathcal{P}, \mathcal{T})$ .*

The first proposition indicates that the intermediate domain  $\mathcal{P}$  shares more semantic similarity with the target domain  $\mathcal{T}$ , thus aligning the intermediate domain is much easier than the direct alignment of source and target domains. This proposition provides a compromised solution when facing an extremely huge domain gap or when there is rare data for aligning these domains.

**Proposition 2** (*Reducing target classification error*). *Aligning the target domain  $\mathcal{T}$  to the intermediate domain proxy  $\mathcal{P}_\lambda$  can reduce the discrepancy distance between the source and target domain  $\text{disc}_{\mathcal{L}}(\mathcal{S}, \mathcal{T})$ , which in turn reduces the error of the classifier  $\epsilon_{\mathcal{T}}$  on the target domain.*



**Fig. 3** Illustration of the proposed method. We first collect dense prototypes from source domains and use them to construct the intermediate proxy pool. Features in this pool are then employed to reconstruct the target domain, forming intermediate reconstructions. After that, the intermediate proxies are adopted as learning guidance for fast feature alignment of target and intermediate domains.

Besides the first proposition, the other crucial issue is that aligning the intermediate domain and target domain should in turn reduce the gap between the source and target domains. Thus we could regard the intermediate domain as a substitution during the optimization. Please refer to the Appendix for detailed proofs.

## 4 Approach

### 4.1 Dense Reconstruction for Cross-domain Few-shot Learning

**Problem Formulation.** Given the source and target domain  $\mathcal{D}_s$  and  $\mathcal{D}_t$ , the distribution  $\mathcal{N}(\cdot)$  of  $\mathcal{D}_s$  and  $\mathcal{D}_t$  are in different semantic space, *i.e.*,  $\mathcal{N}(\mathcal{D}_s) \approx \mathcal{N}(\mathcal{D}_t)$ . The large-scale labeled images of base classes  $C_{base}$  are available in the source domain, *i.e.*,  $\mathcal{D}_s = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}_{i=1}^L$ , where  $L$  is the number of images in  $\mathcal{D}_s$ . For  $N$ -way  $K$ -shot problems, the target domain  $\mathcal{T}$  includes two parts: a support dataset  $\mathcal{T}^s = \{(\mathbf{x}_i^{ts}, \mathbf{y}_i^{ts})\}_{i=1}^{K \times N}$  with a few labeled samples and a unknown query dataset  $\mathcal{T}^q = \{(\mathbf{x}_i^{tq}, \mathbf{y}_i^{tq})\}_{i=1}^M$  for inference. Here  $N, K$  denotes the number of classes and images in each class in  $\mathcal{T}^s$ . The label space of source domain  $\mathcal{D}_s$  is disjoint from target domain  $\mathcal{T}$ , *i.e.*,  $C_{base} \cap C_{novel} = \phi$ . The optimization objective follows the conventional FSL problems, which aim to learn a generalized embedding from base classes and then transfer it to target domains with few-shot data. The optimized parameters are typically composed of two major components: 1) one backbone

network  $f_\theta$  to encode the images into dense feature maps; 2) one classifier  $g_V$  to predict probabilities for each category. Thus we have

$$\mathcal{L}_{ce}(\theta, \mathcal{V}) = \mathbb{E}_{(\mathbf{x}_i^o, \mathbf{y}_i^o) \sim \mathcal{D}_o} [\mathbf{y}_i \log(g_V(\text{Pool}(f_\theta(\mathbf{x}_i^o))))], \quad (3)$$

$$o \in \{s, t\}.$$

**Dense Reconstructions for CDFSL.** The conventional learning schemes in Eq. (3) construct classifiers following a prototypical trend (Snell et al. 2017), *i.e.*,  $\mathbf{V} = \sum_{i=1}^k \sum_{j=1}^{WH} (f_\theta(\mathbf{x}_{i,j}))$  for given  $k$  samples and spatial dimension  $j$ .  $W, H$  are the width and height of the feature map. Although this learning scheme provides satisfactory feature embeddings in common FSL problems, it still suffers two major challenges in cross-domain learning: 1) the spatial information is severely neglected by pooling operations, while for novel categories in the target domain, representing novel objects with only global prototypes are usually difficult; 2) global prototypes provides more domain-specific semantics while losing generalization when serving as materials for reconstruction. Following the reconstruction process in Eq. (1) and prevailing works (Bertinetto et al. 2018; Wertheimer et al. 2021), we here first reconstruct the source domains with prototypes  $\mathbf{V}^s$  without losing its spatial dimension. This reconstruction measurements  $\mathcal{R}_{\mathcal{A} \rightarrow \mathcal{B}}(\cdot)$  by using domain  $\mathcal{A}$  to construct  $\mathcal{B}$  also follows a closed form solution with learnable  $\mathbf{V}^a$ :

$$\mathcal{R}_{\mathcal{A} \rightarrow \mathcal{B}}(f_\theta(\mathbf{x}^b); \mathbf{V}^a) = \frac{\exp(\|\mathbf{W}^a \mathbf{V}_i^a - f_\theta(\mathbf{x}^b)\|^2)}{\sum_{j=0}^{N-1} \exp(\|\mathbf{W}^a \mathbf{V}_j^a - f_\theta(\mathbf{x}^b)\|^2)},$$

(4)

where  $\mathbf{W}^a = f_\theta(\mathbf{x})\mathbf{V}^{a^\top}(\mathbf{V}^a\mathbf{V}^{a^\top} + \lambda\mathbf{I})^{-1}$  denotes the inter-domain mapping weights.  $a, b$  denotes the matrices or vectors belonging to domains  $\mathcal{A}$  and  $\mathcal{B}$  respectively. We thus form a reconstruction  $\mathcal{R}_{\mathcal{S} \rightarrow \mathcal{S}}(\cdot)$  by using source domain bases to construct itself as pretraining to get generalized embedding. After this pretraining stage in only source domains, the densely formed visual prototypes  $\mathbf{V}$  can be constructed and the network parameters are initialized for subsequent cross-domain learning. Note that the reconstruction process for prototypes mainly follows Wertheimer et al. (2021) to optimize the few-shot process in a differentiable manner.

## 4.2 Intermediate Domain Construction

**Intermediate Proxies Generation.** Starting from Eq. (4), we then have a densely formed visual prototypes  $\mathbf{V} \in \mathbb{R}^{WH \times D \times C_{base}}$ , where  $C_{base}, D$  denotes the number of base classes and feature dimensions. The major concern is what we asked in Q1 in Section 3, *i.e.*, how to choose reconstructed materials for the intermediate domain without additional costs? Using base pooling prototypes  $\mathbb{R}^{WH \times D \times C_{base}}$  with such high dimension for reconstruction would lead to huge computation costs and inferior optimization for the closed-form equation in Eq. (1). Hence we use the K-means algorithm to cluster the dense prototypes into spatial feature pool  $\mathcal{U}$  for intermediate domain construction  $\mathcal{P}$ , resulting in a cluster mapping  $\mathcal{M} : \mathbb{R}^{WH \times D \times C_{base}} \rightarrow \mathbb{R}^{WH \times D}$ , as in Fig. 3. This clustering process also regularizes different semantic classes with the unified reconstruction materials. Different from prevailing FSL efforts, the intermediate proxy pool  $\mathcal{U} = \{\mathbf{U}_i\}_{i=1}^{WH}$  depicts a generalized representation of each visual pattern, thus is inherent to reconstruct further novel categories with large domain gaps. By replacing the generalized learnt  $\mathbf{U}$  with  $\mathbf{C}$  that only for specific samples, for each category  $n \in [1, N]$ , we have

$$\mathbf{P}^i = \widehat{\mathbf{W}}^i \mathbf{U} = \mathbf{T}^i \mathbf{U}^\top (\mathbf{U} \mathbf{U}^\top + \lambda \mathbf{I})^{-1} \mathbf{U} \in \mathbb{R}^{KWH \times D}, \quad (5)$$

where  $\mathbf{T}^i$  indicates the  $i$ th class of the target domain features. Eq. (5) means different classes in the novel set share similar reconstruction materials from the proxy pool  $\mathcal{U}$ . We perform reconstruction for each category independently to obtain its respective intermediate domain. This observation leads us to an interesting manner to further tune these proxies and make them fully adapted to target domains, *i.e.*, the constructed proxy  $\mathbf{P}^i$  highly corresponds with the supported features  $\mathbf{T}^i$  for class  $C_i$ . Hence we use the standard cross entropy

to retain the reconstructed proxies in the target domain semantic space with  $C_{novel} = N$  categories:

$$\mathcal{L}_{proxy}(\theta) = \mathbb{E}_{(\mathbf{x}_i^{ts}, \mathbf{y}_i^{ts}) \sim \mathcal{D}_t} [\mathbf{y}_i^{ts} \log(\mathcal{R}_{\mathcal{P} \rightarrow \mathcal{T}}(f_\theta(\mathbf{x}), \mathbf{P}_i))], \quad (6)$$

where  $\mathcal{R}_{\mathcal{P} \rightarrow \mathcal{T}}$  denotes the reconstruction function from proxies  $\mathcal{P}$  to target domains  $\mathcal{T}$ . Note that during this process, the proxies are not learnable matrices and can only be updated indirectly by the change of extracted source prototypes  $\check{\mathbf{V}}^s$ , and then be re-mapped by  $\{\check{\mathbf{U}}\}_i = \mathcal{M}(\check{\mathbf{V}}^s)$ .

## 4.3 Domain Alignment with Intermediate Proxies

After constructing intermediate proxies distributed between source and target domains, one intuitive idea is to align target and intermediate domains  $(\mathcal{P}, \mathcal{T})$  instead of the direct alignment of  $(\mathcal{S}, \mathcal{T})$ . Beyond this consideration, we observed that due to the extremely small size of training samples, this compromised alignment by using intermediate domains still leads to catastrophic overfitting on very few target samples. To overcome this, here we propose to transform the feature statistics beyond tuning the whole network.

**Decomposing Batch Normalization.** Conventional BN layers (Ioffe and Szegedy 2015) are typically decomposed into two stages, statistic normalization and affine transformations. Given an extracted feature map  $\mathbf{F} \in \mathbb{R}^{B \times H \times W \times D}$  with the batch-wise dimension, the mean and variances are calculated along the channel size

$$\mu_{\text{BN}} = \frac{1}{B \times H \times W} \sum_{b=1}^B \sum_{r=1}^{H \times W} \mathbf{F}_{b,c,r}, \quad (7)$$

and the channel-wise variance is computed as

$$\sigma_{\text{BN}}^2 = \frac{1}{B \times H \times W} \sum_{b=1}^B \sum_{r=1}^{H \times W} (\mathbf{F}_{b,c,r} - \mu_{\text{BN}})^2. \quad (8)$$

Thus the feature normalization in Eq. (9) indicates the **low-order statistics** (Maria Carlucci et al. 2017; Wang et al. 2019), which are mainly decided by the historical statistics and current running status. While Eq. (10) forms a **high-order** affine transformation to adjust the distribution shape with the scaling factor  $\gamma$  and shifting factor  $\beta$ .

$$\mathbf{F}_{\text{BN}} = \frac{\mathbf{F} - \mu_{\text{BN}}}{\sqrt{\sigma_{\text{BN}}^2 + \epsilon}}, \quad (9)$$

$$\mathbf{F}_{\text{aff}} = \gamma \mathbf{F}_{\text{BN}} + \beta. \quad (10)$$

**Feature Transformation with Intermediate Domain Proxies.** Dozens of research efforts (Li et al. 2016) have demonstrated the strong correlation between visual styles and feature statistics and several pioneer works (Zhang et al. 2020b) focus on the BN optimization in domain adaptation problems. Inspired by these early explorations, here we resort to the intermediate proxies for the fast alignment of multiple domains. First, the low-order statistics in the normalization layers are updated by a static momentum, *i.e.*, updating the same ratio of samples during different training phases. Here we argue to construct a dynamic momentum function that is gradually increased during the training phase.

$$\tilde{\mu}_t = (1 - \mathcal{G}^\alpha(t)) \cdot \tilde{\mu}_{t-1} + \mathcal{G}^\alpha(t) \cdot \mu_t, \quad (11)$$

$$\tilde{\sigma}_t^2 = (1 - \mathcal{G}^\alpha(t)) \cdot \tilde{\sigma}_{t-1}^2 + \mathcal{G}^\alpha(t) \cdot \sigma_t^2, \quad (12)$$

where  $\mathcal{G}^\alpha(t) = 1/(1 + \exp(-t/\alpha))$ ,  $\alpha$  is used to control the scale of weighting function.  $\tilde{\mu}_t, \tilde{\sigma}_t$  denotes the updated mean and variances for time step  $t$ . Thus in the first training stages, the models tend to use more source statistics for representation stability.

For high-order affine transformations, we resort to aligning the target and intermediate domains by Kullback-Leibler divergences. This alignment only works on the learnable scaling factor  $\gamma$  and shifting factor  $\beta$ , fast adapting the distribution to a proper shape, as in Fig. 3. We hence conduct this constraint on each target support samples  $\mathbf{x}^{ts}$  and its corresponding intermediate proxies with  $\mathcal{R}_{\mathcal{P} \rightarrow \mathcal{T}}(\cdot)$ :

$$\begin{aligned} \mathcal{L}_{align}(\cdot; \beta, \gamma) &= \mathbb{D}_{K-L}(\mathcal{R}(\mathbf{P}, \mathbf{V}^t) \| \mathcal{R}(f_\theta(\mathbf{x}^{ts}; \beta, \gamma), \mathbf{V}^t)) \\ &= \mathcal{R}(f_\theta(\mathbf{x}, \mathbf{V}^t) \log \mathcal{R}(\mathbf{P}, \mathbf{V}^t) + \\ &\quad \mathcal{R}(\mathbf{P}, \mathbf{V}^t) \log \mathcal{R}(f_\theta(\mathbf{x}^{ts}; \beta, \gamma), \mathbf{V}^t). \end{aligned} \quad (13)$$

where  $\mathbf{V}^t \in \mathbb{R}^{WH \times D \times C_{novel}}$  denotes the target learnable prototypes.

#### 4.4 Model Optimization

**Learning Objective.** Besides the pretraining stage on source domains, the target domain learning objective is composed of three constraints: 1) semantic constraints for intermediate proxies  $\mathcal{L}_{proxy}$ ; 2) standard cross-entropy loss for target domain finetuning  $\mathcal{L}_{tar}$  with  $\mathcal{R}_{\mathcal{T} \rightarrow \mathcal{T}}(\cdot)$ :

$$\mathcal{L}_{tar}(\cdot; \theta) = \mathbb{E}_{(\mathbf{x}^{ts}, \mathbf{y}^{ts}) \sim \mathcal{D}_t} [\mathbf{y}^{ts} \log(\mathcal{R}(f_\theta(\mathbf{x}^{ts}), \mathbf{V}^t))], \quad (14)$$

where  $\mathbf{x}_i^{ts}$  and  $\mathbf{y}_i^{ts}$  denote the target domain support set sample and its label respectively; 3) intermediate

---

#### Algorithm 1: Cross-domain Few-shot Learning with Intermediate Domain Proxies (IDP)

---

**Input:** Source domain data  $\mathcal{S}$ , target domain support set  $\mathcal{T}^s$ , target domain query set  $\mathcal{T}^q$ .  
**Output:** Backbone network  $f_\theta$ , Classifier  $g_{\mathcal{V}}$ , Predicted results  $\mathbf{S}$ .

- 1 Init. parameters of the backbone network  $\theta$  in  $F_\theta(\cdot)$  with random norm;
- // **Source Domain Pretraining**
- 2 Random Init. source domain densely formed visual prototypes:  $\mathbf{V}^s \in \mathbb{R}^{WH \times D \times C_{base}}$ ;
- 3 **for**  $\forall(\mathbf{x}^s, \mathbf{y}^s) \in \mathcal{S}$  **do**
- 4   Calculate inter-domain mapping weights  
    $\mathbf{W}^s = f_\theta(\mathbf{x})\mathbf{V}^{s\top}(\mathbf{V}^s\mathbf{V}^{s\top} + \lambda\mathbf{I})^{-1}$ ;
- 5   Calculate reconstruction measurements  
    $\mathcal{R}_{\mathcal{S} \rightarrow \mathcal{S}}(f_\theta(\mathbf{x}); \mathbf{V}^s)$  by Eq. 4;
- 6   Calculate cross-entropy loss  $\mathcal{L}_{ce}$  by Eq. 3 and optimizing  $\theta, \mathcal{V} = \arg \min_{\theta, \mathcal{V}} \mathcal{L}_{ce}$ ;
- 7 **end**
- // **Target Domain Finetuning**
- 8 Init. parameters of the backbone network  $\theta$  in  $F_\theta(\cdot)$  with pre-training;
- 9 **for** *fine-tuning time step*  $t$  and  $\forall(\mathbf{x}^{ts}, \mathbf{y}^{ts}) \in \mathcal{T}^s$  **do**
- 10   Calculate cross-entropy loss for target domain finetuning  $\mathcal{L}_{tar}$ ;
- 11   Cluster the dense prototypes  $\mathbf{V}^s$  into spatial feature pool  $\mathcal{U}$ ;
- 12   Reconstruct intermediate proxies  $\mathbf{P}^i$  using generalized learnt  $\mathbf{U}$  by Eq. 5;
- 13   Calculate cross-entropy loss for reconstructed proxies  $\mathcal{L}_{proxy}$  by Eq. 6;
- 14   Update low-order statistics  $\tilde{\mu}_t, \tilde{\sigma}_t^2$  by Eq. 11 and Eq. 12;
- 15   Calculate reconstruction measurements  
    $\mathcal{R}_{\mathcal{P} \rightarrow \mathcal{T}}(\mathbf{P}; \mathbf{V}^t)$  by Eq. 4;
- 16   Calculate intermediate proxies constraint loss  
    $\mathcal{L}_{align}$  by Eq. 13;
- 17   Gather total loss  $\mathcal{L}_{sum}$  by Eq. 15 and optimizing  
    $\theta, \gamma, \beta = \arg \min_{\theta, \gamma, \beta} \mathcal{L}_{sum}$ ;
- 18 **end**
- // **Target Domain Querying**
- 19 **for**  $\forall(\mathbf{x}^{tq}, \mathbf{y}^{tq}) \in \mathcal{T}^q$  **do**
- 20   Calculate reconstruction measurements  
    $\mathcal{R}_{\mathcal{T} \rightarrow \mathcal{T}}(f_\theta(\mathbf{x}^{tq}); \mathbf{V}^t)$  by Eq. 4;
- 21   Predicting the probability of each category  $\mathbf{S}$  using  $\mathcal{R}_{\mathcal{T} \rightarrow \mathcal{T}}$ ;
- 22 **end**
- 23 **return** Predicted results  $\mathbf{S}$  of  $N$  classes

---

alignment loss for feature transformations  $\mathcal{L}_{align}$ . The overall learning objective has the form:

$$\mathcal{L}_{sum} = w_t \mathcal{L}_{tar}(\cdot; \theta) + w_p \mathcal{L}_{proxy}(\cdot; \theta) + w_a \mathcal{L}_{align}(\cdot; \gamma, \beta). \quad (15)$$

We empirically set balanced weights  $w_{t,p,a}$  as 1, which already shows satisfactory performance despite its simplicity. Note that in the evaluation phase, we do not rely on the intermediate proxies and only use the backbone networks  $f_\theta$  with target domain prototypes  $\mathbf{V}^t$  using measurements in Eq. (4). Besides, following previous



**Table 1** Comparisons with state-of-the-art models on CDFSL benchmark dataset. The first and second best values on each dataset are highlighted in bold and underlined. \*: Finetuning using the same optimization as Ours. †: re-trained using ResNet-12 as backbone, others using ResNet-10.

| 1-shot                                     | ISIC              | EuroSAT           | CropDisease       | ChestX            | Car               | CUB               | Plantae           | Places            |
|--|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| GNN (Garcia and Bruna 2018)                | 32.02±0.66        | 63.69±1.03        | 64.48±1.08        | 22.00±0.46        | 31.79±0.51        | 45.69±0.68        | 35.60±0.56        | 53.10±0.80        |
| FWT (Tseng et al. 2020)                    | 31.58±0.67        | 62.36±1.05        | 66.36±1.04        | 22.04±0.44        | 31.61±0.53        | 47.47±0.75        | 35.95±0.58        | 55.77±0.79        |
| LRP (Sun et al. 2021)                      | 30.94±0.30        | 54.99±0.50        | 59.23±0.50        | 22.11±0.20        | 32.78±0.39        | 48.29±0.51        | 37.49±0.43        | 54.83±0.56        |
| AFA (Hu and Ma 2022)                       | 33.21±0.30        | 63.12±0.50        | 67.61±0.50        | 22.92±0.20        | 34.25±0.40        | 46.86±0.50        | 36.76±0.40        | 54.04±0.60        |
| STARTUP (Phoo and Hariharan 2020)          | 32.66±0.60        | 63.88±0.84        | 75.93±0.80        | 23.09±0.43        | -                 | -                 | -                 | -                 |
| TPN-ATA (Wang and Deng 2021a)              | 33.21±0.40        | 61.35±0.50        | 67.47±0.50        | 22.10±0.20        | 33.61±0.40        | 45.00±0.50        | 34.42±0.40        | 53.57±0.50        |
| FRN <sup>†</sup> (Wertheimer et al. 2021)  | 33.73±0.62        | 63.80±0.91        | 71.93±0.85        | 22.52±0.40        | 32.37±0.58        | <b>51.76±0.80</b> | <u>42.37±0.73</u> | <b>56.92±0.84</b> |
| FRN* <sup>†</sup> (Wertheimer et al. 2021) | 33.38±0.58        | 60.25±0.81        | 70.09±0.82        | 22.53±0.38        | 33.08±0.57        | 44.95±0.74        | 36.45±0.65        | 50.84±0.75        |
| ConFT (Das et al. 2021)                    | <u>34.47±0.60</u> | 64.79±0.80        | 69.71±0.90        | <b>23.31±0.4</b>  | <b>39.11±0.77</b> | 45.57±0.76        | 43.09±0.78        | 49.97±0.86        |
| KT (Li et al. 2023)                        | 34.06±0.77        | <u>66.43±0.93</u> | <u>73.10±0.87</u> | 22.68±0.60        | -                 | -                 | -                 | -                 |
| IDP (Ours)                                 | <b>35.94±0.53</b> | <b>71.60±0.67</b> | <b>83.85±0.60</b> | <u>23.11±0.33</u> | <u>38.46±0.53</u> | <u>49.70±0.61</u> | <b>44.39±0.61</b> | 54.07±0.63        |
| 5-shot                                     | ISIC              | EuroSAT           | CropDisease       | ChestX            | Car               | CUB               | Plantae           | Places            |
| GNN (Garcia and Bruna 2018)                | 43.94±0.67        | 83.64±0.77        | 87.96±0.67        | 25.27±0.46        | 44.28±0.63        | 62.25±0.65        | 52.53±0.59        | 70.84±0.65        |
| FWT (Tseng et al. 2020)                    | 43.17±0.70        | 83.01±0.79        | 87.11±0.67        | 25.18±0.45        | 44.90±0.64        | 66.98±0.68        | 53.85±0.62        | 73.94±0.67        |
| LRP (Sun et al. 2021)                      | 44.14±0.40        | 77.14±0.40        | 86.15±0.40        | 24.53±0.30        | 46.20±0.46        | 64.44±0.48        | 54.46±0.46        | 74.45±0.47        |
| AFA (Hu and Ma 2022)                       | 46.01±0.40        | 85.58±0.40        | 88.06±0.30        | 25.02±0.20        | 49.28±0.50        | 68.25±0.50        | 54.26±0.40        | 76.21±0.50        |
| FT-All (Guo et al. 2020)                   | 48.11±0.64        | 79.08±0.61        | 89.25±0.51        | 25.97±0.41        | 52.08±0.74        | 64.14±0.77        | 59.27±0.70        | 70.06±0.74        |
| STARTUP (Phoo and Hariharan 2020)          | 47.22±0.61        | 82.29±0.60        | 93.02±0.45        | 26.94±0.94        | -                 | -                 | -                 | -                 |
| FRN <sup>†</sup> (Wertheimer et al. 2021)  | 47.41±0.59        | 80.77±0.60        | 91.93±0.46        | 26.77±0.40        | 49.78±0.68        | <b>73.06±0.72</b> | 61.04±0.74        | 73.65±0.71        |
| FRN* <sup>†</sup> (Wertheimer et al. 2021) | 47.17±0.58        | 80.52±0.62        | 90.68±0.47        | 25.18±0.41        | 50.92±0.70        | 67.29±0.71        | 56.07±0.73        | 69.71±0.69        |
| ATA-FT (Wang and Deng 2021a)               | 49.79±0.40        | <u>89.64±0.30</u> | <u>95.44±0.20</u> | 25.08±0.20        | 54.28±0.50        | 69.83±0.50        | 58.08±0.40        | <u>76.64±0.40</u> |
| NSAE (Liang et al. 2021)                   | <u>54.05±0.63</u> | 83.96±0.57        | 93.14±0.47        | <u>27.10±0.44</u> | 54.91±0.74        | 68.51±0.76        | 59.55±0.74        | 71.02±0.72        |
| BSR (Liu et al. 2020)                      | <b>54.42±0.66</b> | 80.89±0.61        | 92.17±0.45        | 26.84±0.44        | 57.49±0.72        | 69.38±0.76        | 61.07±0.76        | 71.09±0.68        |
| ConFT (Das et al. 2021)                    | 50.79±0.60        | 81.52±0.60        | 90.90±0.60        | <b>27.50±0.50</b> | <u>61.53±0.75</u> | 70.53±0.75        | 62.54±0.76        | 72.09±0.68        |
| ConFeSS (Das et al. 2022)                  | 48.85±0.29        | 84.65±0.38        | 88.88±0.51        | 27.09±0.24        | -                 | -                 | -                 | -                 |
| KT (Li et al. 2023)                        | 46.37±0.77        | 82.53±0.66        | 89.53±0.58        | 26.79±0.61        | -                 | -                 | -                 | -                 |
| IDP (Ours)                                 | 53.36±0.50        | <b>91.08±0.41</b> | <b>96.89±0.28</b> | 26.87±0.34        | <b>62.76±0.56</b> | <u>72.92±0.58</u> | <b>69.10±0.56</b> | <b>78.08±0.55</b> |

cross-domain few-shot learning methods (Tseng et al. 2020; Wang and Deng 2021b; Hu and Ma 2022), we also incorporate the GNN model (Garcia and Bruna 2018) as our classifiers. GNN learns the joint relationship of support and query samples to predict the probability of each sample belonging to each class  $S$  based on the target domain reconstruction metric  $\mathcal{R}(f_{\theta}(\mathbf{x}); \mathbf{V}^t)$ .

Alg. 1 shows the training and inference details of the proposed approach. We divide the whole learning scheme into three stages. 1) We first pre-train our model on an annotated source domain dataset, such as *mini-ImageNet* (Vinyals et al. 2016), using each pair of data  $(\mathbf{x}^s, \mathbf{y}^s)$ . 2) we fine-tuned our model using the support set of the target domain  $(\mathbf{x}^{ts}, \mathbf{y}^{ts})$  to adapt it to the target domain gradually. 3) we use the optimized model to classify the samples of the target domain query set  $\mathbf{x}^{tq}, \mathbf{y}^{tq}$ . In this manner, our training scheme shows two distinctive advantages compared to vanilla implementations: i) our approach does not rely on additional target domain unlabeled data or source domain data; ii) Our final inference network is lightweight and does not rely

on additional constructed intermediate domains. These two advantages indicate our “free lunch” implementation without any data or computation burden during the domain alignment process.

**Discussions.** The key challenge for cross-domain few-shot learning is the data scarcity and huge domain gap. Our proposed IDP benefits from two major points to solve this challenge.

1. Intermediate domains to reconstruct target domain content: Our approach does not directly utilize the source domain features and align this feature to the target domain, which is typically applied in the previous domain adaptation works (Robey et al. 2021; Kang et al. 2018; Zhang et al. 2019). Instead, our approach reconstructs the source domain features through an intermediate proxy, which still leans towards the target domain in terms of content. Additionally, our method allows effective control of the influence of the source domain on the intermediate domain by adjusting the source domain feature pool

sizes, thereby suppressing the risk of the model overfitting during the adaptation.

2. Optimizing normalization statistics other than CNN weights: Direct aligning with the intermediate domain might also lead to overfitting when there are only extremely few samples used for training. Thus we proposed to optimize the feature transformation statistics, *i.e.*, high-order and low-order statistics in the normalization layers, instead of the direct optimization on holistic network parameters. In this manner, the feature distribution can be globally transformed to align with the intermediate domain and the relative semantic relationships of the pretrained features from the source domains can be maintained, leading to the fast global alignment with extremely limited data.

## 5 Experiments

### 5.1 Experiment Setting

**Datasets.** Following the benchmark setting in CDFSL, we use the miniImageNet (Vinyals et al. 2016) training set as the source domain. Mini-ImageNet is a subset of ILSVRC-2012 (Deng et al. 2009), and its training set part has 64 classes, each containing 100 natural images collected from the Internet. In addition, we use eight target domain datasets to respond to real scenarios. For diverse levels of cross-domain learning, we follow BSCD-FSL (Guo et al. 2020) which includes CropDisease, EuroSAT, ISIC, and ChestX datasets. For natural images in other CDFSL methods, we follow the dataset split of Tseng et al. (2020), which includes Car (Krause et al. 2013), CUB (Wah et al. 2011), Plantae (Van Horn et al. 2018) and Places (Zhou et al. 2017) datasets.

**Evaluation Protocol.** To make fair comparisons, we follow benchmark protocol (Guo et al. 2020), which involves validating the performance of the classifiers by simulating 600 independent 5-way  $k$ -shot tasks in the target domain. Since large shots can be easily learned by supervised learning, we conduct experiments with  $k \in \{1, 5\}$ . For each task, we randomly select 5 categories from all categories in the target domain dataset and, within each category, we randomly select  $k$  images for the support set and 16 images for the query set. For each task, we fine-tune the pre-trained model on the support set and evaluate its performance on the query set. We repeat this process 600 times for each experiment setting, resulting in 600 fine-tuned and evaluated models. Average classification accuracy and its 95% confidence interval on the query set are reported in accordance with the benchmark evaluations.

**Implementation Details.** To make a fair comparison with existing works (Guo et al. 2020; Phoo and Har-iharan 2020; Liang et al. 2021), in all experiments we use ResNet-10 backbone network with SGD optimizer. For pretraining, we set the learning rate to 0.05 and the batch size to 64 for 350 epochs. We then conduct meta-finetuning on each target domain for 50 epochs with a learning rate of 0.01. We found that setting the prototype number of each class  $\mathbf{V}_i^t$  to 20 is sufficient to obtain satisfactory results. To ensure that each pixel on the feature map has a sufficient field of perception, we set  $W = H = 5$ , thus the input image resolution is scaled to  $160 \times 160$ . Following previous works (Tseng et al. 2020; Wang and Deng 2021b; Hu and Ma 2022), we also used a meta-trained lightweight GNN (Garcia and Bruna 2018) as final classifiers to learn the relationship of few-shot samples to obtain the final results. As the test phase required less graphics memory and could be executed on a lower-performance GPU, our model was implemented in the PyTorch (Paszke et al. 2019) framework on a single NVIDIA GTX3090 GPU.

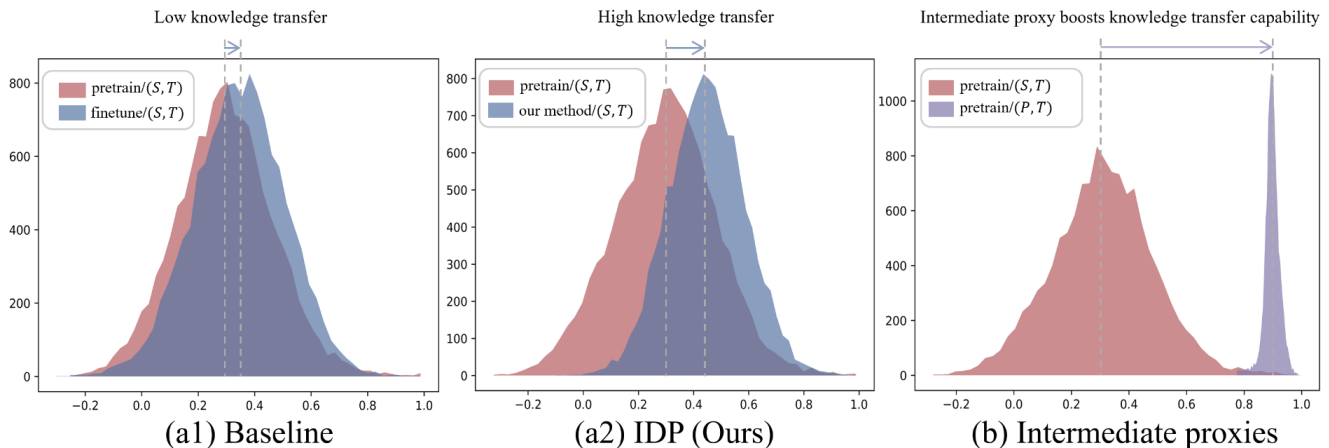
### 5.2 Comparison with State-of-The-Art

**Benchmarking on Domain with Diverse Levels.** We first conduct comparisons on the most widely-used BSCD-FSL benchmark (Guo et al. 2020) with the state-of-the-art models. Tab. 1 exhibits the results of different levels of domain transfer, *i.e.*, a gradual decrease of visual features from CropDisease and EuroSAT to ISIC and ChestX shared with the source domain. For fair comparisons, we also extend the FRN (Wertheimer et al. 2021) on this cross-domain few-shot setting with the prototypical networks (Snell et al. 2017) on cross-domain classes and we finetuned FRN with the identical hyper-parameters and optimizer, noted as FRN\*. Identical to our method, FRN\* adopts the SGD optimizer with a learning rate of 0.01 and performs meta-finetuning on each target domain for 50 epochs. From Tab. 1, the finetuning on FRN leads to an inferior performance than FRN, indicating that the FRN representations are easy to overfit on few-shot samples, especially on these datasets with huge domain gaps. Note that FRN methods are built upon the ResNet-12 backbone while others are using the lightweight ResNet-10 backbones. Our method on average achieves 53.63% and 67.05% on 5-way 1-shot and 5-shot settings respectively, outperforming all the listed CDFSL competitors significantly, including ConFT (Das et al. 2021), ConFeSS (Das et al. 2022), and KT (Li et al. 2023), and our proposed IDP leads a new state-of-the-art.

**Benchmarking on Natural Images.** Besides the diverse domain benchmarking, the other line of work

**Table 2** Comparisons with state-of-the-art models using the 5-way random-shot setting on Meta-Dataset benchmark. The performances are evaluated using official codes and the best values on each set are highlighted in bold.

| Test Dataset | KT Li et al. (2023) | LDP-Net Zhou et al. (2023) | TSA Li et al. (2022b) | IDP(ours)    |
|--------------|---------------------|----------------------------|-----------------------|--------------|
| ILSVRC       | 62.51               | 59.86                      | <b>80.37</b>          | 77.42        |
| Omniglot     | 79.34               | 90.15                      | 93.01                 | <b>98.64</b> |
| Aircraft     | 44.48               | 60.94                      | 61.88                 | <b>75.14</b> |
| Birds        | 55.86               | 57.51                      | <b>84.80</b>          | 71.28        |
| Textures     | 65.59               | 65.35                      | 75.76                 | <b>78.82</b> |
| Quick Draw   | 71.97               | 83.98                      | 78.71                 | <b>87.64</b> |
| Fungi        | 50.82               | 45.84                      | <b>69.98</b>          | 68.46        |
| VGG Flower   | 78.84               | 82.83                      | 91.63                 | <b>96.28</b> |
| Traffic Sign | 56.42               | 86.04                      | 75.07                 | <b>88.04</b> |
| MSCOCO       | 58.71               | 62.17                      | 72.90                 | <b>76.16</b> |
| Avg.         | 62.45               | 69.47                      | 78.41                 | <b>81.79</b> |

**Fig. 4** Frequency distribution of randomly sampled distances between feature pairs from different domains (*Higher values indicates better domain alignments.*). a1) and a2): Domain-transferring ability of **Baseline** and **Ours**. b): **Intermediate domain proxies** boost knowledge-transferring capabilities.**Table 3** Ablation studies of 5-way K-shot learning of different modules.

| Method                  | EuroSAT          |                  | Places           |                  |
|-------------------------|------------------|------------------|------------------|------------------|
|                         | 1-shot           | 5-shot           | 1-shot           | 5-shot           |
| GNN (base)              | 63.69±1.0        | 83.64±0.8        | 53.10±0.8        | 70.84±0.7        |
| + $\mathcal{L}_{tar}$   | 66.82±0.6        | 87.59±0.4        | 53.39±0.7        | 74.34±0.6        |
| + $\mathcal{L}_{proxy}$ | 68.95±0.7        | 90.11±0.5        | 53.73±0.7        | 77.37±0.6        |
| + $\mathcal{L}_{align}$ | <b>71.60±0.7</b> | <b>91.08±0.4</b> | <b>54.07±0.6</b> | <b>78.08±0.6</b> |

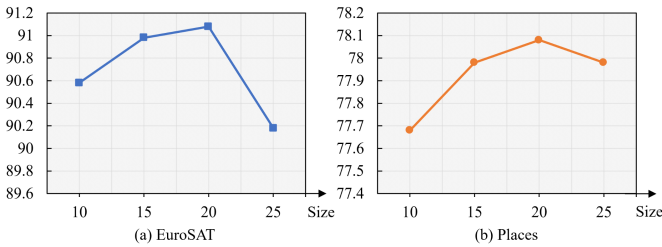
**Table 4** Effects of different optimization combinations on 5-way K-shot learning performance.  $BN_l$  and  $BN_h$  indicates the low-order and high-order statistics.

| Optim.     | EuroSAT            |                    | Places             |                    |
|------------|--------------------|--------------------|--------------------|--------------------|
|            | 1-shot             | 5-shot             | 1-shot             | 5-shot             |
| $f_\theta$ | 69.54±0.7          | 88.74±0.4          | 52.30±0.7          | 76.83±0.6          |
| $BN_l$     | ✓ 70.66±0.7        | ✓ 90.76±0.4        | ✓ 53.83±0.7        | ✓ 77.78±0.6        |
| $BN_h$     | ✓ 69.18±0.7        | ✓ 88.84±0.5        | ✓ 53.72±0.7        | ✓ 77.32±0.6        |
| ✓          | ✓ <b>71.60±0.7</b> | ✓ <b>91.08±0.4</b> | ✓ <b>54.07±0.6</b> | ✓ <b>78.08±0.6</b> |

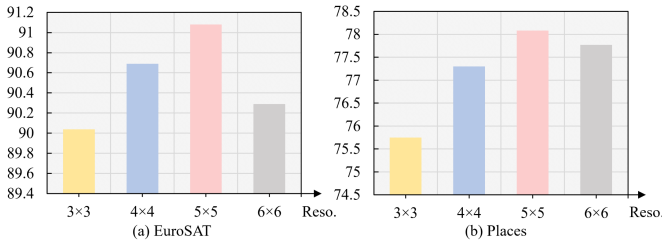
focuses on natural images but with different data distributions. Tab. 1 presents the comparisons on natural

images, *e.g.*, cars and birds, with representative state-of-the-art methods including NSAE (Liang et al. 2021), ConFT (Das et al. 2022). Similar to the CDFSL benchmark, following the class split of Tseng et al. (2020), we also conduct experiments on FRN and finetuned FRN (FRN\*) on these natural domain images. Note that FRN (Wertheimer et al. 2021) adopted ResNet-12 as backbones while others used ResNet-10. FRN shows better performance in a 1-shot setting when there are fewer domain gaps, *e.g.*, CUB datasets. When more samples are available (*i.e.*, 5-shot), our method surpasses FRN by over 6.33% on average. Besides, it can be observed that our method shows leading results to prevailing methods under this natural domain setting. Among them, our method outperforms the second-place method ConFT (Das et al. 2022) by over 4% on average for the 5-way 5-shot setting, indicating the strong generalization capabilities of our method.

**Extensions on Meta-Dataset.** Compared to the prevailing datasets, the Meta-Dataset (Triantafillou et al. 2019) is a large-scale benchmark consisting of ILSVRC-2012 (Deng et al. 2009) as the training set and ten in-



**Fig. 5** The effect of different size of class prototype  $\mathbf{V}_i^t$ . All experiments are conducted under 5-way 5-shot conditions, and the vertical coordinates indicate the performance of our method.



**Fig. 6** The effect of resolution of the feature map. All experiments were conducted under 5-way 5-shot conditions, and the vertical coordinates indicate the performance of our method.

dividual datasets as the testing set. Additionally, the Meta-Dataset benchmark (Triantafillou et al. 2019) introduces varying samples for each class, determined based on the distribution of real-world data. For each task, we randomly select 5 categories from all categories in the test dataset. Following Triantafillou et al. (2019), the sample size for each category’s support set is a random number in  $[1, 100]$ , while the number of query samples is fixed at 10, as all categories hold equal importance. We computed the average accuracy and 95% confidence interval for 600 independently sampled tasks while keeping the other implementation details consistent with the CDFSL benchmark. The results of our method are presented in Tab. 2. We select three state-of-the-art methods for comparison, *i.e.*, KT (Li et al. 2023), LDP-Net (Zhou et al. 2023), and TSA (Li et al. 2022b). We conduct the 5-way random-shot experiments with their open-source code for fair comparisons. Compared with these methods, our IDP maintains its leading position on the Meta-Dataset benchmark and achieves clear improvements on multiple subsets, indicating its powerful generalization ability to adapt to real-world scenarios.

### 5.3 Performance Analyses

**Effect of Different Components.** To evaluate the effectiveness of our method, we perform ablation studies in Tab. 3. In the first row, we show the performance of the baseline model GNN,  $\mathcal{L}_{tar}$  represents the addition measurements to obtain the predicted values. It

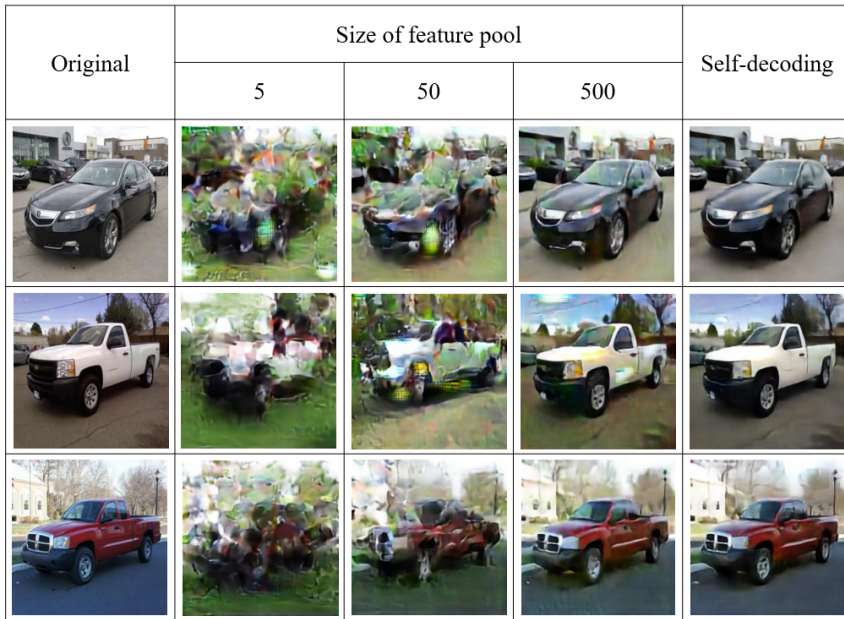
can be found that this metric effectively improves the discriminative ability of the model. Further, we use the spatial feature pool  $\mathcal{U}$  to reconstruct the target domain support set samples to obtain intermediate proxies  $\mathbf{P}$ . After that, we optimize the classifier with the cross-entropy loss  $\mathcal{L}_{proxy}$  of these intermediate proxies with labels as shown in the third row, which improves the cross-domain generalizability of the model. We eventually add intermediate proxy alignment loss  $\mathcal{L}_{align}$  to further improve the final performance.

**Variant of Alignment Loss Optimization for Intermediate Proxies.** In Tab. 4, we compare the effects of optimizing different components of the network, where  $f_\theta$  represents the optimization of the whole network parameters,  $\text{BN}_l$  represents the tuning of the statistics of the BN layer using intermediate proxies, and  $\text{BN}_h$  represents the optimization of the learnable scaling and shifting parameters of the BN layer. Comparing the first two rows we can observe that the proposed strategy for adjusting the BN layer statistics is effective. Further, we find that optimizing the entire network parameters using alignment loss  $\mathcal{L}_{align}$  in the third row overfits the model to the intermediate proxies, which impairs the performance. Finally, we use alignment loss to optimize only the higher-order learnable parameters of the BN layer  $\text{BN}_h$ , achieving the highest performance in the last row.

**Effect of Intermediate Proxy Alignment Loss on Distribution.** In Fig. 4, we show the difference between the distribution of the source and target domains (a1) without alignment loss  $\mathcal{L}_{align}$  and (a2) with alignment loss. We conduct the following steps to calculate the alignment scores: a) Randomly sampling pairs of data from different domains; b) Calculating the Euclidean distance of their L2 normalized representations; c) Plotting the frequency distribution of distances within small intervals. It can be observed that the addition of the alignment loss significantly reduces the source and target domain metrics, *i.e.*, the difference distribution is closer to 1. This shows that our approach can better transfer the knowledge learned by the model during the pre-training phase on the source domain to the target domain. 4(b) shows the difference between the distribution of target domain samples and intermediate proxies, and it can be seen that the intermediate proxies are closer to the target domain, hence it is easier to align them.

**Effect of Prototype Size.** In Fig. 5, we exhibit how different sizes of class prototype  $\mathbf{V}_i^t$  affect the performance of our method. It can be observed that on the one hand if the prototype size is too small then the representation of the categories is insufficient, on the other





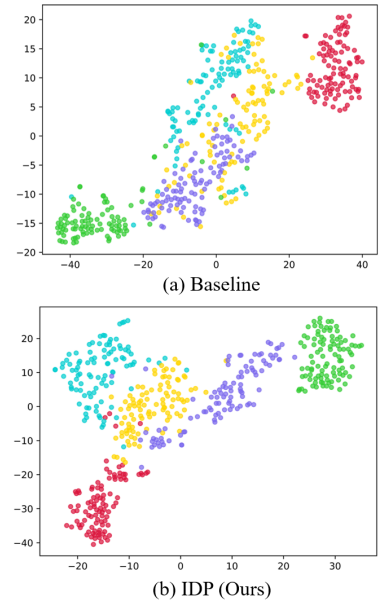
**Fig. 7** Visualization of reconstruction using different source domain feature pool sizes.

hand, a large prototype size reduces the discriminative ability of the model.

**Effect of Feature Resolution for Reconstruction.** In Fig. 6, we present how the resolution of the feature map affects the performance of our method. It can be observed that the size of the image region corresponding to each vector on the feature map affects the retention of details by the network. Smaller scales ( $3 \times 3$ ) of the network features usually lead to the coarse observation of local details, while larger scales ( $7 \times 7$ ) lead to an inferior focus on the global reception field.

**Extensions on Intra-domain Gap.** To evaluate the effectiveness of our method, the extreme scenarios for cross-domain learning is “intra-domain”, *i.e.*, evaluated on the same miniImageNet dataset (Vinyals et al. 2016). We compare with the recent few-shot learning method on the mini-ImageNet dataset. Our proposed method uses the ResNet-10 network as backbones but still achieves good performance in this intra-domain setting, as in Tab. 5, which indicates the strong generalization ability when facing fewer domain gaps.

**Computational Efficiency.** Our proposed method shows similar computational costs compared to the baseline methods (Garcia and Bruna 2018). Benefiting from the optimization scheme, the intermediate domain proxies are dropped during the inference stage, and our method does not rely on many additional network parameters. The detailed inference time and computational costs are presented in Tab. 6. With the additional 8.5% costs, our method improves the baseline by a large margin, *e.g.*, 43.96% to 53.36% on the ISIC dataset.



**Fig. 8** t-SNE comparisons of 5-way classification on target domains (EuroSAT).

**Table 5** Comparisons with state-of-the-art models on mini-ImageNet benchmark dataset. The best values on each set are highlighted in bold. †: using ResNet-12 backbones. \*: using lightweight ResNet-10 backbone.

| Method                             | 5-way 1-shot      | 5-way 5-shot      |
|------------------------------------|-------------------|-------------------|
| FEAT† (Ye et al. 2020)             | 66.78±0.20        | 82.05±0.14        |
| H-OT† (Guo et al. 2022)            | 65.63±0.32        | 82.87±0.43        |
| SAPENet† (Huang and Choi 2023)     | 66.41±0.20        | 82.76±0.14        |
| MatchingNet* (Vinyals et al. 2016) | 58.76±0.61        | 72.53±0.69        |
| RelationNet* (Sung et al. 2018)    | 58.64±0.85        | 73.78±0.64        |
| GNN* (Garcia and Bruna 2018)       | 66.32±0.80        | 81.98±0.55        |
| <b>IDP(Ours)*</b>                  | <b>67.16±0.71</b> | <b>84.64±0.46</b> |

**Table 6** Comparison of computational efficiency with representative methods.

| Method                        | GFLOPS       | Time (ms)   |
|-------------------------------|--------------|-------------|
| GNN (Garcia and Bruna 2018)   | 189.0        | 14.6        |
| GNN-ATA (Wang and Deng 2021a) | 196.1        | 15.9        |
| TPN-ATA (Wang and Deng 2021a) | 211.6        | 17.3        |
| KT (Li et al. 2023)           | 214.1        | 19.6        |
| <b>IDP (Ours)</b>             | <b>205.2</b> | <b>19.3</b> |

## 5.4 Visualization and Explanations

**Effect of Source Domain Feature Pool.** In Fig. 7, we show the relationship between the number of feature pool sizes and the reconstruction. It can be observed that as the pool of features involved in the reconstruction becomes larger, the intermediate proxies are able to reconstruct the target domain samples more

clearly. Conversely, when the proxy pool is relatively small ( $\leq 50$ ), the reconstructed intermediate proxies are more ambiguous and show more of the source domain style. This indicates that the intermediate domain reconstruction is *controllable* when changing the reconstruction materials in the stored feature pool.

**Target Domain Category Embedding.** We visualize the target domain embeddings in the EuroSAT dataset in Fig. 8 using t-SNE. Fig. 8 shows that our methods generate few intra-class differences and larger inter-class differences compared to the baseline. This indicates after domain adaptation, our proposed method retains a stronger discriminative capability with only a few given samples.

## 6 Conclusions and Limitations

In this paper, we start from a different view to revisit the problem of cross-domain few-shot learning. Prevailing research efforts mainly focus on the generalized representation of feature learning while neglecting the fast domain alignment with these few samples. Toward this end, we propose to reconstruct an intermediate domain using source embeddings and use the reconstructed domain proxies to develop a fast domain transformation technique with normalization layers. Despite its superior performance on public CDFSL benchmarks, our proposed method still relies on dense feature reconstructions, which may limit the extension of our work on segmentation and dense estimation vision tasks, which we leave for our future exploration.

**Acknowledgements** This work is partially supported by grants from the National Natural Science Foundation of China under contracts No. 62132002, No. 62202010, and in part by the Fundamental Research Funds for the Central Universities.

## Appendix

### A. Additional Details of the Empirical Study

#### A.1 Implementation Details

To conduct the empirical study, we first organize two “sub-Domains”, domain  $\mathcal{A}$  and domain  $\mathcal{B}$ . Both “sub-Domain” are sampled from *mini-ImageNet* (Vinyals et al. 2016) and contain a series of stylistically distinct and visually similar image classes. We give more examples of domain  $\mathcal{A}$  and domain  $\mathcal{B}$  in Fig. 9 and 10, where it can be observed that since objects in domain  $\mathcal{A}$  are often located in the jungle or on grass, they are visually greenish in color; in contrast, objects in domain

$\mathcal{B}$  are often located underwater and are visually bluish in color. The impact of feature base  $\{\mathbf{C}_i\}_{i=1}^n$  on domain reconstruction can be inferred by observing the performance of the reconstructed intermediate domain proxies  $\mathcal{P}$  in terms of style and content.

For image level reconstruction  $\text{img} \rightarrow \mathbf{r.img}$ , we first resize the image to  $224 \times 224$  and slice the image into blocks of size  $7 \times 7$ , each with a length and width of 32. feature bases  $\{\mathbf{C}_i\}_{i=1}^n$ . We flatten these image blocks into vectors, which are used as our feature bases  $\{\mathbf{C}_i\}_{i=1}^n \in \mathbb{R}^{n \times 1024}$ . It is worth noting that since the reconstruction process is independent of the spatial location of the pixels, the flattening operation does not affect the results of the image block reconstruction. We then solve for the intermediate agent  $\mathcal{P}$  according to Eq. 2.

For feature level reconstruction  $\text{feat} \rightarrow \mathbf{r.feat}$ , we still resize the image to 224, which will result in an output feature map size of  $7 \times 7$  for backbone network  $f(\theta)$ . We use the pixels on the feature map as feature bases  $\{\mathbf{C}_i\}_{i=1}^n \in \mathbb{R}^{n \times d}$  and perform a reconstruction process similar to the image level reconstruction described above, where  $d$  is the output feature dimension of the  $f(\theta)$ .

#### A.2 Visualization Method

For image level reconstruction  $\text{img} \rightarrow \mathbf{r.img}$ , we reshape the reconstructed vectors into image blocks and stitch these image blocks into a new image according to their spatial locations. For feature level reconstruction  $\text{feat} \rightarrow \mathbf{r.feat}$ , we obtain feature maps as the reconstruction results. However, since these feature maps cannot be directly visualized, we propose to utilize a decoder to convert them into images. Table 7 illustrates the architectural specifications of the decoder, which can be seen as a mirrored version of the ResNet10 backbone network. It consists of Deconv blocks, each Deconv block containing an upsampling function, a convolution operator, and a ReLU activation. We train the decoder to decode the original images from the feature maps, which are generated by the encoder. Finally, we utilize the decoder to visualize the intermediate representations.

#### A.3 More Visualization Results

In Fig. 11, we visualize more intermediate proxies for image-level reconstructions. We can observe a strong relationship between the number of image blocks involved in the reconstruction and the reconstruction results. When reconstructing the target image using only

**Table 7** The architecture specifications of the decoder modules in ResNet10. We insert a BatchNorm layer behind each Deconv layer.

| Module   | Specifications                                     |
|----------|--|
| ResNet10 | 3×3 Deconv-ReLU, 256 filters, stride 2, padding 1  |
|          | 3×3 Deconv-ReLU, 128 filters, stride 2, padding 1  |
|          | 3×3 Deconv-ReLU, 64 filters, stride 2, padding 1   |
|          | 3×3 Deconv-ReLU, 32 filters, stride 2, padding 1   |
|          | 3×3 Deconv-Sigmoid, 3 filters, stride 2, padding 1 |

one image block, *i.e.*, the first column of Fig. 11, the intermediate proxy is simply a concatenation of different brightness and contrast combinations of that image block. As the number of image blocks involved in the reconstruction increases, the intermediate agents behave from blurred to clear and eventually very close to the target image.

## B. Proof of Proposition 1

**Proposition 1** (*High semantic similarity*). *By controlling the ridge regression regular term  $\lambda$ , the semantic similarity between the intermediary domain proxy  $\mathcal{P}_\lambda$  and the target domain  $\mathcal{T}$  is larger than that between the source domain  $\mathcal{S}$  and the target domain  $\mathcal{T}$ . Their inter-domain discrepancy distance satisfies the following relationship:  $\exists \lambda$ , s.t.  $\text{disc}_{\mathcal{L}}(\mathcal{S}, \mathcal{T}) > \text{disc}_{\mathcal{L}}(\mathcal{P}, \mathcal{T})$*

*Proof.* We first recall the formula for ridge regression:

$$\widehat{\mathbf{W}} = \arg \min_{\mathbf{W}} \|\mathbf{T} - \mathbf{W}\mathbf{U}\|^2 + \lambda \|\mathbf{W}\|^2, \quad (16)$$

where  $\mathbf{U} \in \mathbb{R}^{n \times d}$  denote the generalized representation of each source domain visual pattern and  $\mathbf{T} \in \mathbb{R}^{r \times d}$  denote the target domain embeddings required to be reconstructed. The hyper-parameter  $\lambda$  is used to balance the regularization of the  $\ell_2$ -norm. As stated in Section 4.2, we begin by resizing the feature pool through clustering mapping  $\mathcal{M} : \mathbb{R}^{n \times d} \rightarrow \mathbb{R}^{r \times d}$ . This operation aligns the size of  $\mathbf{U}$  with  $\mathbf{T}$  and sets the dimension of  $\mathbf{W}$  to  $r \times r$ .

Next, we define  $\mathbf{F}(\lambda) = \|\mathbf{T} - \mathbf{P}\|^2 - \|\mathbf{T} - \mathbf{U}\|^2$  as the difference from the  $\mathbf{T}$  to the  $\mathbf{P}$  and from  $\mathbf{T}$  to  $\mathbf{U}$ . Since  $\widehat{\mathbf{W}} = \mathbf{T}\mathbf{U}^\top(\mathbf{U}\mathbf{U}^\top + \lambda\mathbf{I})^{-1}$  is the ridge regression closed-form solution,  $\mathbf{F}$  can be rewritten as:

$$\mathbf{F}(\lambda) = \|\mathbf{T} - \mathbf{T}\mathbf{U}^\top(\mathbf{U}\mathbf{U}^\top + r\mathbf{I})^{-1}\|^2 - \|\mathbf{T} - \mathbf{U}\|^2. \quad (17)$$

We then solve for the partial derivative of  $\mathbf{F}$  with respect to  $r$ .

$$\frac{\partial \mathbf{F}(\lambda)}{\partial r} = 2\text{tr}(\mathbf{H}(\mathbf{T}^\top - \mathbf{H}\mathbf{U}\mathbf{T}^\top)\mathbf{T}\mathbf{U}^\top\mathbf{H}), \quad (18)$$

where  $\text{tr}$  denotes the trace of the matrix and  $\mathbf{H} = (\mathbf{U}\mathbf{U}^\top + r\mathbf{I})^{-1}$  in order to simplify the expression. It can be observed that the monotonicity of  $\mathbf{F}$  is determined by both the regular term  $r$  and the distribution of the data. Therefore, we discuss it by situation:

**If  $\lambda$  is equal to 0.** According to the property of ridge regression, we can obtain that for any  $\mathbf{W}^* \in \mathbb{R}^{n \times n}$ , the target domain embeddings  $\mathbf{T}$  satisfies

$$\|\mathbf{T} - \widehat{\mathbf{W}}\mathbf{U}\|^2 + \lambda \|\widehat{\mathbf{W}}\|^2 \leq \|\mathbf{T}_i - \mathbf{W}^*\mathbf{U}\|^2 + \lambda \|\mathbf{W}^*\|^2, \quad (19)$$

where  $\widehat{\mathbf{W}} = \mathbf{T}\mathbf{U}^\top(\mathbf{U}\mathbf{U}^\top + \lambda\mathbf{I})^{-1}$  is the ridge regression closed-form solution. Assuming  $\mathbf{T} \neq \mathbf{U}$ , we then substitute the identity matrix  $\mathbf{I} \in \{0, 1\}^{r \times r}$  for  $\mathbf{W}^*$  in this equation, and obtain:

$$\|\mathbf{T} - \widehat{\mathbf{W}}\mathbf{U}\|^2 + \lambda \|\widehat{\mathbf{W}}\|^2 < \|\mathbf{T} - \mathbf{U}\|^2 + \lambda \|\mathbf{I}\|^2. \quad (20)$$

Bringing  $\lambda = 0$  into the Eq. (20), we can prove that  $\mathbf{F} < 0$ .

**If  $\lambda > 0$  and  $\mathbf{F}$  is monotonically increasing.** Data noise may causes  $\|\mathbf{T} - \mathbf{W}\mathbf{U}\|^2$  to be monotonically increasing, since  $r = 0$  with  $\mathbf{F}(r) < 0$  holds, there must exist a  $\lambda = \lambda'$  near 0 for  $\mathbf{F}(\lambda) < 0$  to hold.

**If  $\lambda > 0$  and  $\mathbf{F}$  is decreases first**, there will exist  $\lambda > \lambda'$  such that  $\mathbf{F}(\lambda) < 0$ .

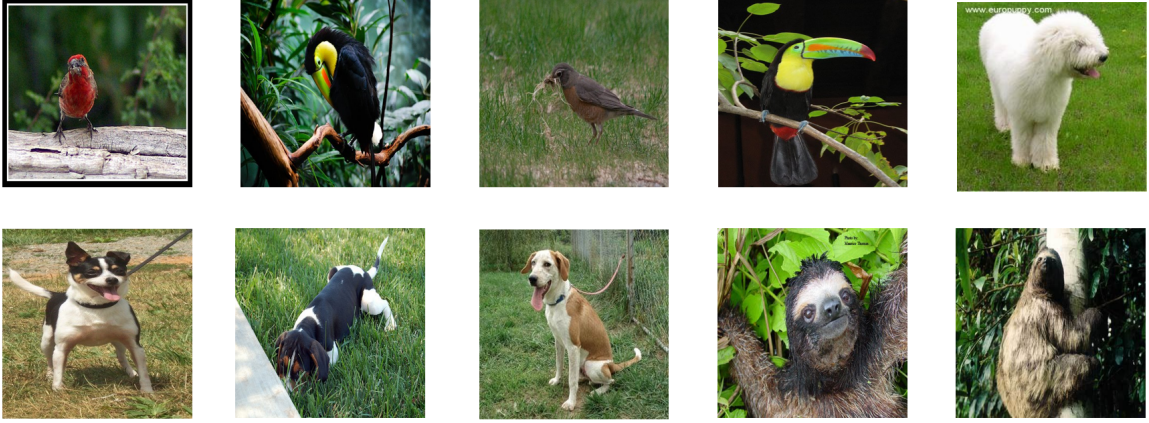
According to Definition 1, we can conclude that Proposition 1 holds.  $\square$

## C. Proof of Proposition 2

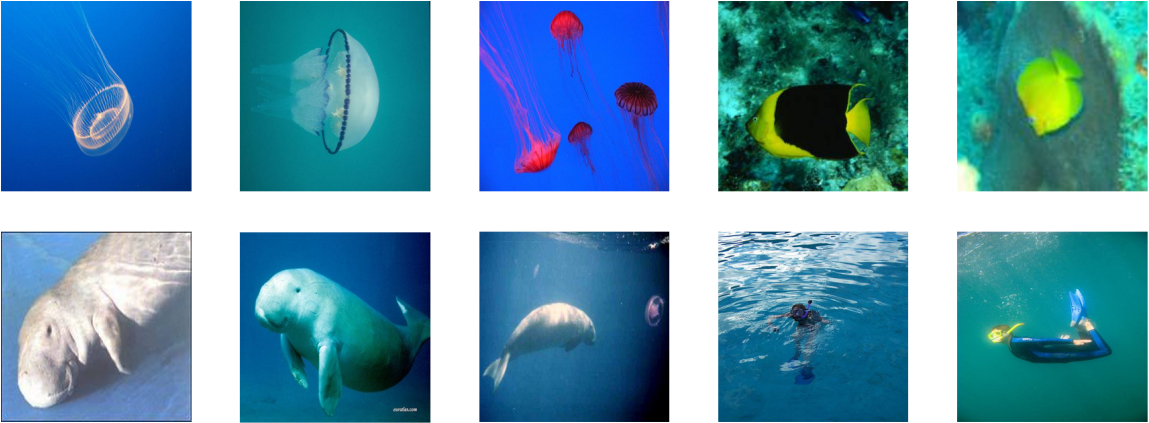
**Proposition 2** (*Reducing target classification error*). *Aligning the target domain  $\mathcal{T}$  to the intermediate domain proxy  $\mathcal{P}_\lambda$  can reduce the discrepancy distance between the source and target domain  $\text{disc}_{\mathcal{L}}(\mathcal{S}, \mathcal{T})$ , which in turn reduces the error of the classifier  $\epsilon_{\mathcal{T}}$  on the target domain.*

*Proof.* We start by defining the symbols. Specifically, we denote  $l$  as the class labeling function, with  $l_{\mathcal{S}}$  representing the function for the source domain and  $l_{\mathcal{T}}$  representing the function for the target domain. Consider a hypothesis set  $H = \{h\}_i$ , and let  $h_{\mathcal{S}}^* \in \arg \min_{h \in H} \mathcal{L}_{\mathcal{S}}^c(h, l_{\mathcal{S}})$  and  $h_{\mathcal{T}}^* \in \arg \min_{h \in H} \mathcal{L}_{\mathcal{T}}^c(h, l_{\mathcal{T}})$  be the classifiers that minimize the empirical risk on the source dataset  $\mathcal{S}$  and the target dataset  $\mathcal{T}$ , respectively (Zhang et al. 2020c). The hypothesis error of the target domain classifier can be defined as  $\epsilon_{\mathcal{T}} = \mathcal{L}_{\mathcal{T}}^c(h, l_{\mathcal{T}}) - \mathcal{L}_{\mathcal{T}}^c(h_{\mathcal{T}}^*, l_{\mathcal{T}})$ . Since our difference distance loss function  $\text{disc}_{\mathcal{L}}$  is symmetric and obeys the triangle inequality, according to





**Fig. 9** More illustration of objects in domain  $\mathcal{A}$ . Domain  $\mathcal{A}$  consists of partial *mini-ImageNet* dataset categories, including birds, dogs, and sloths, which are objects in the jungle or on the grass.



**Fig. 10** More illustration of objects in domain  $\mathcal{B}$ . Domain  $\mathcal{B}$  consists of partial *mini-ImageNet* dataset categories, including jellyfish, manatee, and butterfly fish, which are underwater objects.

domain adaptation theory (Mansour et al. 2009; Ben-David et al. 2010) for any hypothesis  $h \in H$ , the following holds:

$$\mathcal{L}_{\mathcal{T}}^c(h, l_{\mathcal{T}}) \leq \mathcal{L}_{\mathcal{T}}^c(h_{\mathcal{T}}^*, l_{\mathcal{T}}) + \mathcal{L}_{\mathcal{S}}^c(h, h_{\mathcal{S}}^*) + \text{disc}_{\mathcal{L}^a}(\mathcal{S}, \mathcal{T}) + \mathcal{L}_{\mathcal{S}}^c(h_{\mathcal{S}}^*, h_{\mathcal{T}}^*). \quad (21)$$

This inequality can be transformed into

$$\epsilon_{\mathcal{T}} \leq \mathcal{L}_{\mathcal{S}}^c(h, h_{\mathcal{S}}^*) + \text{disc}_{\mathcal{L}^a}(\mathcal{S}, \mathcal{T}) + \mathcal{L}_{\mathcal{S}}^c(h_{\mathcal{S}}^*, h_{\mathcal{T}}^*). \quad (22)$$

We observe that the hypothesis error with respect to the target domain  $\epsilon_{\mathcal{T}}$  is linked to the average loss of source classification  $\mathcal{L}_{\mathcal{S}}^c(h, h_{\mathcal{S}}^*)$ , the discrepancy distance  $\text{disc}_{\mathcal{L}^a}(\mathcal{S}, \mathcal{T})$ , and the average loss between the best intra-class hypotheses  $\mathcal{L}_{\mathcal{S}}^c(h_{\mathcal{S}}^*, h_{\mathcal{T}}^*)$ .

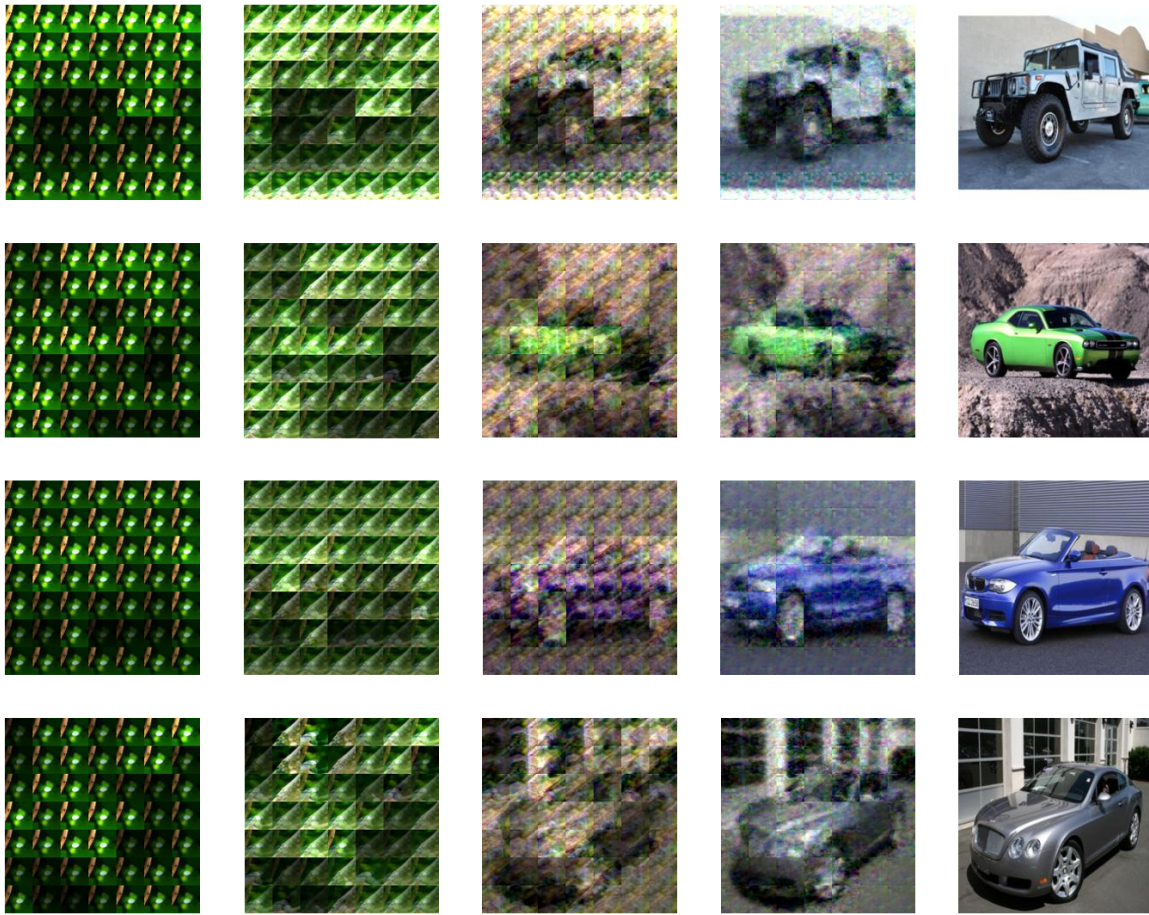
Since our method follows a two-stage training format,  $\mathcal{L}_{\mathcal{S}}^c(h, h_{\mathcal{S}}^*)$  remains constant after pre-training. Furthermore, in order to achieve successful domain adaptation, it is reasonable for the optimal classifiers  $h_{\mathcal{S}}^*$  and  $h_{\mathcal{T}}^*$  in the source and target domains, respectively, to exhibit low inconsistency in semantic prediction, as

measured by  $\mathcal{L}_{\mathcal{S}}^c(h_{\mathcal{S}}^*, h_{\mathcal{T}}^*)$ . With our proposed alignment loss  $\mathcal{L}_{\text{align}}$ , the model will force for samples in the target domain  $\mathcal{T}$  to extract more features that express the style of the source domain  $\mathcal{S}$ , i.e., reduce  $\text{disc}_{\mathcal{L}^a}(\mathcal{S}, \mathcal{T})$  to  $\text{disc}_{\mathcal{L}^a}(\mathcal{S}, \mathcal{P})$  as the optimization proceeds. According to the conclusion of Proposition 1, this will reduce the target domain classifier error  $\epsilon_{\mathcal{T}}$ .  $\square$

## References

- Ben-David S, Blitzer J, Crammer K, Kulesza A, Pereira F, Vaughan JW (2010) A theory of learning from different domains. *Machine learning* 79:151–175
- Bertinetto L, Henriques JF, Torr P, Vedaldi A (2018) Meta-learning with differentiable closed-form solvers. In: *International Conference on Learning Representations*
- Chen WY, Liu YC, Kira Z, Wang YCF, Huang JB (2019) A closer look at few-shot classification. *arXiv preprint arXiv:190404232*
- Chen Y, Rosenfeld E, Sellke M, Ma T, Risteski A (2022) Iterative feature matching: Toward provable domain generalization with logarithmic environments. *Advances in Neural Information Processing Systems* 35:1725–1736





**Fig. 11** Illustration of image level reconstruction  $\text{img} \rightarrow \text{r. img}$ . The source domain is set to domain  $\mathcal{A}$ . The number of image blocks involved in the reconstruction from left to right in the first four columns is  $n = \{1, 5, 20, 50\}$ , respectively. The rightmost column is the target image.

Cui S, Wang S, Zhuo J, Su C, Huang Q, Tian Q (2020) Gradually vanishing bridge for adversarial domain adaptation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 12455–12464

Dai Y, Liu J, Sun Y, Tong Z, Zhang C, Duan LY (2021) Idm: An intermediate domain module for domain adaptive person re-id. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 11864–11874

Das D, Yun S, Porikli F (2022) Confess: A framework for single source cross-domain few-shot learning. In: International Conference on Learning Representations

Das R, Wang YX, Moura JM (2021) On the importance of distractors for few-shot classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 9030–9040

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

Doersch C, Gupta A, Zisserman A (2020) Crosstransformers: spatially-aware few-shot transfer. *Advances in Neural Information Processing Systems* 33:21981–21993

Fei-Fei L, Fergus R, Perona P (2006) One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence* 28(4):594–611

Finn C, Abbeel P, Levine S (2017) Model-agnostic meta-learning for fast adaptation of deep networks. In: International

conference on machine learning, PMLR, pp 1126–1135

Ganin Y, Ustinova E, Ajakan H, Germain P, Larochelle H, Laviolette F, Marchand M, Lempitsky V (2016) Domain-adversarial training of neural networks. *The journal of machine learning research* 17(1):2096–2030

Garcia V, Bruna J (2018) Few-shot learning with graph neural networks. In: 6th International Conference on Learning Representations, ICLR 2018

Gatys LA, Ecker AS, Bethge M (2015) A neural algorithm of artistic style. *arXiv preprint arXiv:150806576*

Gidaris S, Bursuc A, Komodakis N, Pérez P, Cord M (2019) Boosting few-shot visual learning with self-supervision. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 8059–8068

Gong B, Shi Y, Sha F, Grauman K (2012) Geodesic flow kernel for unsupervised domain adaptation. In: 2012 IEEE conference on computer vision and pattern recognition, IEEE, pp 2066–2073

Gopalan R, Li R, Chellappa R (2013) Unsupervised adaptation across domain shifts by generating intermediate data representations. *IEEE transactions on pattern analysis and machine intelligence* 36(11):2288–2302

Guo D, Tian L, Zhao H, Zhou M, Zha H (2022) Adaptive distribution calibration for few-shot learning with hierarchical optimal transport. *Advances in neural information*

- processing systems 35:6996–7010
- Guo Y, Codella NC, Karlinsky L, Codella JV, Smith JR, Saenko K, Rosing T, Feris R (2020) A broader study of cross-domain few-shot learning. In: European conference on computer vision, Springer, pp 124–141
- He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
- Helber P, Bischke B, Dengel A, Borth D (2019) Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 12(7):2217–2226
- Hoerl AE, Kennard RW (1970) Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* 12(1):55–67
- Hu Y, Ma AJ (2022) Adversarial feature augmentation for cross-domain few-shot classification. In: *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XX*, Springer, pp 20–37
- Huang X, Choi SH (2023) Sapenet: Self-attention based prototype enhancement network for few-shot learning. *Pattern Recognition* 135:109170
- Ioffe S, Szegedy C (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: *International conference on machine learning*, pmlr, pp 448–456
- Kang G, Zheng L, Yan Y, Yang Y (2018) Deep adversarial attention alignment for unsupervised domain adaptation: the benefit of target expectation maximization. In: *Proceedings of the European conference on computer vision (ECCV)*, pp 401–416
- Kemker R, McClure M, Abitino A, Hayes TL, Kanan C (2018) Measuring catastrophic forgetting in neural networks. In: *Thirty-Second AAAI Conference on Artificial Intelligence*
- Krause J, Stark M, Deng J, Fei-Fei L (2013) 3d object representations for fine-grained categorization. In: *Proceedings of the IEEE international conference on computer vision workshops*, pp 554–561
- Lake BM, Salakhutdinov R, Tenenbaum JB (2015) Human-level concept learning through probabilistic program induction. *Science* 350(6266):1332–1338
- Li P, Gong S, Wang C, Fu Y (2022a) Ranking distance calibration for cross-domain few-shot learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 9099–9108
- Li P, Liu F, Jiao L, Li S, Li L, Liu X, Huang X (2023) Knowledge transduction for cross-domain few-shot learning. *Pattern Recognition* 141:109652
- Li WH, Liu X, Bilen H (2022b) Cross-domain few-shot learning with task-specific adapters. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 7161–7170
- Li Y, Wang N, Shi J, Liu J, Hou X (2016) Revisiting batch normalization for practical domain adaptation. *arXiv preprint arXiv:160304779*
- Li Z, Zhou F, Chen F, Li H (2017) Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:170709835*
- Liang H, Zhang Q, Dai P, Lu J (2021) Boosting the generalization capability in cross-domain few-shot learning via noise-enhanced supervised autoencoder. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp 9424–9434
- Liu B, Zhao Z, Li Z, Jiang J, Guo Y, Ye J (2020) Feature transformation ensemble model with batch spectral regularization for cross-domain few-shot classification. *arXiv preprint arXiv:200508463*
- Luo X, Wei L, Wen L, Yang J, Xie L, Xu Z, Tian Q (2021) Rectifying the shortcut learning of background for few-shot learning. *Advances in Neural Information Processing Systems* 34:13073–13085
- Mairal J, Bach F, Ponce J, Sapiro G (2010) Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research* 11(1)
- Mansour Y, Mohri M, Rostamizadeh A (2009) Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:09023430*
- Maria Carlucci F, Porzi L, Caputo B, Ricci E, Rota Bulò S (2017) Autodial: Automatic domain alignment layers. In: *Proceedings of the IEEE international conference on computer vision*, pp 5067–5075
- McCloskey M, Cohen NJ (1989) Catastrophic interference in connectionist networks: The sequential learning problem. In: *Psychology of learning and motivation*, vol 24, Elsevier, pp 109–165
- Miller EG, Matsakis NE, Viola PA (2000) Learning from one example through shared densities on transforms. In: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, IEEE, vol 1, pp 464–471
- Nichol A, Achiam J, Schulman J (2018) On first-order meta-learning algorithms. *arXiv preprint arXiv:180302999*
- Oreshkin B, Rodríguez López P, Lacoste A (2018) Tadaml: Task dependent adaptive metric for improved few-shot learning. *Advances in neural information processing systems* 31
- Pan SJ, Tsang IW, Kwok JT, Yang Q (2010) Domain adaptation via transfer component analysis. *IEEE transactions on neural networks* 22(2):199–210
- Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, et al. (2019) Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32
- Phoo CP, Hariharan B (2020) Self-training for few-shot transfer across extreme task differences. *arXiv preprint arXiv:201007734*
- Rajpurkar P, Irvin J, Zhu K, Yang B, Mehta H, Duan T, Ding D, Bagul A, Langlotz C, Shpanskaya K, et al. (2017) Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:171105225*
- Rizve MN, Khan S, Khan FS, Shah M (2021) Exploring complementary strengths of invariant and equivariant representations for few-shot learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 10836–10846
- Robey A, Pappas GJ, Hassani H (2021) Model-based domain generalization. *Advances in Neural Information Processing Systems* 34:20210–20229
- Rusu AA, Rao D, Sygnowski J, Vinyals O, Pascanu R, Osindero S, Hadsell R (2018) Meta-learning with latent embedding optimization. *arXiv preprint arXiv:180705960*
- Shirekar OK, Singh A, Jamali-Rad H (2023) Self-attention message passing for contrastive few-shot learning. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp 5426–5436
- Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint*

- arXiv:14091556
- Snell J, Swersky K, Zemel R (2017) Prototypical networks for few-shot learning. *Advances in neural information processing systems* 30
- Sun J, Lapuschkin S, Samek W, Zhao Y, Cheung NM, Binder A (2021) Explanation-guided training for cross-domain few-shot classification. In: *2020 25th International Conference on Pattern Recognition (ICPR)*, IEEE, pp 7609–7616
- Sung F, Yang Y, Zhang L, Xiang T, Torr PH, Hospedales TM (2018) Learning to compare: Relation network for few-shot learning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1199–1208
- Triantafillou E, Zhu T, Dumoulin V, Lamblin P, Evcı U, Xu K, Goroshin R, Gelada C, Swersky K, Manzagol PA, et al. (2019) Meta-dataset: A dataset of datasets for learning to learn from few examples. In: *International Conference on Learning Representations*
- Tseng HY, Lee HY, Huang JB, Yang MH (2020) Cross-domain few-shot classification via learned feature-wise transformation. *arXiv preprint arXiv:200108735*
- Tzeng E, Hoffman J, Zhang N, Saenko K, Darrell T (2014) Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:14123474*
- Tzeng E, Hoffman J, Saenko K, Darrell T (2017) Adversarial discriminative domain adaptation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 7167–7176
- Van Horn G, Mac Aodha O, Song Y, Cui Y, Sun C, Shepard A, Adam H, Perona P, Belongie S (2018) The inaturalist species classification and detection dataset. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 8769–8778
- Vinyals O, Blundell C, Lillicrap T, Wierstra D, et al. (2016) Matching networks for one shot learning. *Advances in neural information processing systems* 29
- Vuorio R, Sun SH, Hu H, Lim JJ (2019) Multimodal model-agnostic meta-learning via task-aware modulation. *Advances in Neural Information Processing Systems* 32
- Wah C, Branson S, Welinder P, Perona P, Belongie S (2011) The caltech-ucsd birds-200-2011 dataset
- Wang H, Deng ZH (2021a) Cross-domain few-shot classification via adversarial task augmentation. *arXiv preprint arXiv:210414385*
- Wang H, Deng ZH (2021b) Cross-domain few-shot classification via adversarial task augmentation. In: Zhou ZH (ed) *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization*, pp 1075–1081, main Track
- Wang X, Peng Y, Lu L, Lu Z, Bagheri M, Summers RM (2017) Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 2097–2106
- Wang X, Jin Y, Long M, Wang J, Jordan MI (2019) Transferable normalization: Towards improving transferability of deep neural networks. *Advances in neural information processing systems* 32
- Wei XS, Xu HY, Zhang F, Peng Y, Zhou W (2022) An embarrassingly simple approach to semi-supervised few-shot learning. *Advances in Neural Information Processing Systems* 35:14489–14500
- Wertheimer D, Tang L, Hariharan B (2021) Few-shot classification with feature map reconstruction networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 8012–8021
- Xu J, Luo X, Pan X, Li Y, Pei W, Xu Z (2022a) Alleviating the sample selection bias in few-shot learning by removing projection to the centroid. *Advances in Neural Information Processing Systems* 35:21073–21086
- Xu Y, Wang L, Wang Y, Qin C, Zhang Y, Fu Y (2022b) Memrein: Rein the domain shift for cross-domain few-shot learning. In: *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pp 3636–3642
- Ye HJ, Hu H, Zhan DC, Sha F (2020) Few-shot learning via embedding adaptation with set-to-set functions. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 8808–8817
- Zhang C, Cai Y, Lin G, Shen C (2020a) Deepemd: Few-shot image classification with differentiable earth mover’s distance and structured classifiers. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 12203–12213
- Zhang J, Qi L, Shi Y, Gao Y (2020b) Generalizable semantic segmentation via model-agnostic learning and target-specific normalization. *arXiv preprint arXiv:200312296* 2(3):6
- Zhang Y, Liu T, Long M, Jordan M (2019) Bridging theory and algorithm for domain adaptation. In: *International conference on machine learning*, PMLR, pp 7404–7413
- Zhang Y, Wei Y, Wu Q, Zhao P, Niu S, Huang J, Tan M (2020c) Collaborative unsupervised domain adaptation for medical image diagnosis. *IEEE Transactions on Image Processing* 29:7834–7844
- Zhou B, Lapedriza A, Khosla A, Oliva A, Torralba A (2017) Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence* 40(6):1452–1464
- Zhou F, Wang P, Zhang L, Wei W, Zhang Y (2023) Revisiting prototypical network for cross-domain few-shot learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 20061–20070

## Data Availability Statement

The datasets generated during and/or analyzed during the current research are publicly available in the following references, *i.e.*, ImageNet (Deng et al. 2009) <https://www.image-net.org/>, Stanford Cars (Krause et al. 2013) [https://ai.stanford.edu/~jkrause/cars/car\\_dataset.html](https://ai.stanford.edu/~jkrause/cars/car_dataset.html), CUB-200-2011 (Wah et al. 2011) [https://www.vision.caltech.edu/datasets/cub\\_200\\_2011/](https://www.vision.caltech.edu/datasets/cub_200_2011/), Plantae (Van Horn et al. 2018) [https://github.com/visipedia/inat\\_comp/tree/master/2017](https://github.com/visipedia/inat_comp/tree/master/2017) and Places datasets (Zhou et al. 2017) <http://places.csail.mit.edu/>. Diverse domain benchmarking for CropDisease, EuroSAT, ISIC, and ChestX are included in BSCD-FSL benchmark (Guo et al. 2020) <https://github.com/IBM/cdfsl-benchmark>. The source codes and models corresponding to this study are publicly available.