information

MDPI

Article

Task-Adaptive Multi-Source Representations for Few-Shot Image Recognition †

Ge Liu *, Zhongqiang Zhang and Xiangzhong Fang *

Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China; zhangzhongqiang@sjtu.edu.cn

* Correspondence: liu.ge@sjtu.edu.cn (G.L.); xzfang@sjtu.edu.cn (X.F.)

† This article is a revised and expanded version of a paper entitled: Ge Liu, et al. Learning and Adapting Diverse Representations for Cross-domain Few-shot Learning, which was presented In Proceedings of the IEEE International Conference on Data Mining Workshops, Shanghai, China, 2023.

Abstract: Conventional few-shot learning (FSL) mainly focuses on knowledge transfer from a single source dataset to a recognition scenario with only a few training samples available but still similar to the source domain. In this paper, we consider a more practical FSL setting where multiple semantically different datasets are available to address a wide range of FSL tasks, especially for some recognition scenarios beyond natural images, such as remote sensing and medical imagery. It can be referred to as multi-source cross-domain FSL. To tackle the problem, we propose a two-stage learning scheme, termed learning and adapting multi-source representations (LAMR). In the first stage, we propose a multi-head network to obtain efficient multi-domain representations, where all source domains share the same backbone except for the last parallel projection layers for domain specialization. We train the representations in a multi-task setting where each in-domain classification task is taken by a cosine classifier. In the second stage, considering that instance discrimination and class discrimination are crucial for robust recognition, we propose two contrastive objectives for adapting the pre-trained representations to be task-specialized on the few-shot data. Careful ablation studies verify that LAMR significantly improves representation transferability, showing consistent performance boosts. We also extend LAMR to single-source FSL by introducing a dataset-splitting strategy that equally splits one source dataset into sub-domains. The empirical results show that LAMR can achieve SOTA performance on the BSCD-FSL benchmark and competitive performance on *mini*-ImageNet, highlighting its versatility and effectiveness for FSL of both natural and specific imaging.

Keywords: few-shot learning; image recognition; transfer learning; domain adaptation

Citation: Liu, G.; Zhang, Z.; Fang, X. Task-Adaptive Multi-Source Representations for Few-Shot Image Recognition. *Information* **2024**, *15*, 293. https://doi.org/10.3390/info15060293

Academic Editors: Danilo Avola, Zhikui Chen and Xiaodi Huang

Received: 11 March 2024
Revised: 11 May 2024
Accepted: 16 May 2024
Published: 21 May 2024

1. Introduction

Recent years have witnessed significant progress in computer vision applications thanks to the development of deep learning [1,2] with large-scale annotated data [3]. However, when the deployed domain is specific, the training data may be limited or the labeling cost can be particularly extreme as it must be done by an expert, for example, a doctor in the medical area. To relax the demanding data requirements in deep learning, the emerging topic of few-shot learning (FSL) [4] has received considerable attention and developed as a fundamental research problem in the past few years. With only a few annotated samples per class available, few-shot image recognition aims to efficiently build a classification model for recognizing new classes in an unseen domain.

Directly training a deep recognition model [2] from scratch with only scarce data would intuitively lead to over-fitting collapse [5]. So recent few-shot image recognition is typically addressed in an inductive transfer learning paradigm [6], which aims to improve the learning with limited few-shot data (typically denoted as support set $\mathcal{S}$) using the knowledge in a base set $\mathcal{D}_b$ containing abundant samples. Conventionally, the learning

process is divided into two stages: (1) learning a transferable model from the base dataset $\mathcal{D}_b$, and (2) adapting the pre-trained model to the unseen target few-shot task with $\mathcal{S}$.

Prevailing approaches [7–10] to learning in the first stage are typically based on meta-learning [11]. It learns a meta-model by maximizing the generalization accuracy across a variety of few-shot tasks drawn from the base set, with the goal of transferring meta-knowledge to improve generalization on the unseen domain. The meta-model is shown to hold the promise of fast adaptation [8] and avoiding over-fitting [7]. Although meta-learning provides an elegant solution to FSL, recent studies also indicate that the sophisticated meta-learning algorithms may be unnecessary [12–16]. Instead, the simple representation learning based on supervised cross-entropy loss on the entire dataset could transfer well, and achieve even better performance. Those findings significantly underpin that the essential role of few-shot transfer mainly relies on feature reuse [17] instead of fast adaptation. Other techniques, including self-supervised learning [18] and knowledge distillation [14], have also effectively improved feature transferability. Besides directly leveraging the frozen representation for target FSL, some efforts have also explored the improvements based on task-specific adaptation [19], indicating that proper adaptation may still be necessary [20], especially for cross-domain FSL [21].

However, most of those existing FSL protocols and methods limit their source domains to using only one dataset for pre-training, but many datasets from semantically different domains are indeed available. Besides, a recent benchmark called meta-dataset [22] suggests using multiple source datasets to deal with FSL, but its target datasets for evaluations are still just natural images. In practice, the scenarios for FSL are more likely to come from the specific recognition domains, such as remote sensing [23] and medical imagery [24–26]. In this paper, we aim to address the practical few-shot setting, referred to as multi-source cross-domain few-shot learning. To promote FSL with knowledge from multiple source domains, some methods [27–29] are devoted to learning *universal representations* but still lack effective adaptations. Unlike most prior methods that focus on either representation learning or adaptation on the few-shot data, we address the problem by exploring both aspects: how to effectively learn different generalizable features from multiple source domains and how to use few-shot data to make an efficient adaptation (or deployment) for a wide range of cross-domain FSL scenarios. Therefore, we propose a novel two-stage learning scheme (as illustrated in Figure 1), namely learning and adapting representations (LAMR).

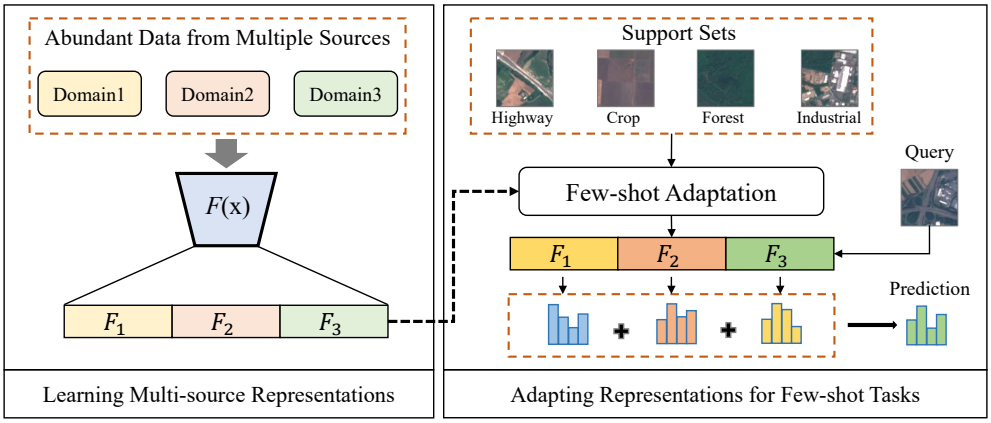


Figure 1. Illustration of our approach. First, we pre-train efficient multi-domain feature representations on the abundant data from semantically different domains. Then, given a few-shot task (such as remote sensing scene recognition), we perform adaptation on the pre-trained multi-domain representations by optimizing domain-specific parameters with few-shot data (the support set).

Concretely, in the first stage, we propose a parameter-efficient multi-head framework for training multi-source representations. Instead of learning a single domain-agnostic embedding, we aim to represent diverse features by constructing separate sub-spaces, each of which corresponds to a specific domain. This is achieved by optimizing multiple in-domain

classification tasks on the multi-head representation spaces with a shared backbone. In this way, our model can preserve information with regard to each domain in a compact network. The representations can then be universal enough to further support generalization to vastly different FSL tasks.

The pre-trained representations are expected to generalize well to the unseen task that is similar to the source domain. However, this is still a challenge if a large domain shift exists between the source and target data, where pre-trained features are less transferable and proper task-specific adaptation on the limited target data becomes necessary. Besides, we consider that instance discrimination and class discrimination are two crucial capabilities for a robust recognition model. To impose the two objectives, we accordingly propose two feature contrastive losses for improving model discrimination towards unseen classes on the few-shot training data. This enables effective task-specific adaptation as the adapted features can be more task-relevant to the target classes. Empirical results show that the adaptation can yield significant performance boosts; the recognition scenario suffers extreme domain shifts, such as remote sensing and medical domains.

In summary, our contributions are as follows:

- We develop a novel two-stage learning scheme, namely learning and adapting representations (LAMR), for vastly addressing cross-domain few-shot learning tasks, especially for recognition scenarios beyond natural images, including remote sensing and medical imagery.
- To achieve multi-source representations, we propose a parameter-efficient multi-head framework, which can further support simple but effective transfer to different downstream FSL tasks.
- To achieve task-specific transfer, we propose a few-shot adaptation method for improving model discrimination towards unseen classes by imposing instance discrimination and class discrimination at the feature level.
- LAMR can achieve state-of-the-art results on cross-domain FSL benchmarks in the multi-source setting.

Compared to the preliminary version of the conference paper [30], this work additionally presents the following novel contents:

- We extend LAMR to single-source FSL by introducing dataset-splitting strategies that equally split one source dataset into sub-domains. The empirical results show that applying simple "random splitting" can improve conventional cosine-similarity-based classifiers in FSL with a fixed single-source data budget. LAMR also achieves superior performance on (single-source) BSCD-FSL benchmark and competitive results on *mini*-ImageNet.
- We conduct more careful ablation studies, which verify that the performance gains come from not only the good transferability of the proposed multi-source representations but also each component in the objectives of few-shot adaptation.
- Discussions and comparisons of more related works, especially for few-shot learning with multi-source domains, are included.
- More feature visualizations and analyses are included. Limitations and future directions are discussed.

The rest of this paper is organized as follows. In Section 2, we briefly review the related works. In Section 3, we formulate the task and present baseline methods. In Section 4, we elaborate on our proposed method. In Sections 5 and 6, we describe the benchmark datasets, implementation details, experiment results, and ablation studies. In Section 7, we draw conclusions, discuss limitations and provide some promising future directions.

2. Related Works

2.1. Few-Shot Learning

Meta-learning [11] is a pioneering approach to addressing few-shot learning [7–10]. The corresponding training regime is namely episodic training, which focuses on mimicking

a target few-shot task style, i.e., "N-way K-shot task". Concretely, this approach trains a meta-model on various "N-way K-shot" tasks (or episodes) sampled from the source dataset, with the goal of steering meta-knowledge that can generalize well in the unseen domain. For instance, MAML [7] optimizes a model-agnostic meta-initialization that can enable fast adaptation to a novel FSL task with only a few fine-tuning steps. Meta-LSTM [8] suggests using an LSTM module as the meta-learner to provide the task-specific update rule for the optimization. Prototypical networks [9] and Matching Networks [10] seek to learn a good metric space capable of directly separating new, unknown classes. Those methods are proven to avoid over-fitting and hold the promise of fast adaptation, as the meta-model is assumed to be the optimal initialization for the different unseen few-shot tasks. Although meta-learning is an elegant solution to the problem of few-shot learning, recent research indicates that it may not be necessary to use complex algorithms. Instead, the simple representation learning [12–16,21,31] based on supervised cross-entropy loss on the entire dataset could transfer well and achieve competitive or even better performance. Those findings significantly highlight that the essential role of few-shot transfer mainly relies on source-feature reuse [17] instead of fast adaptation. Furthermore, to make the feature representations more generalizable and transferable, other techniques, including self-supervised learning [18], knowledge distillation [14,32], saliency-guided attention [33], and contrastive learning [34] have also been proved to effectively improve the performance and enhance model discrimination on the novel categories.

The methods discussed above assume only one source dataset is used for pre-training, but many datasets collected from semantically different domains can be available in the machine learning community. To promote few-shot learning with knowledge from multiple domains, learning *Universal Representations* [21,27–29] and feature selection [21,27,28] are explored in the literature. Concretely, the simplest way [21] to achieve such representations is to train separate feature extractors for each available domain. SUR [27] and URT [28] obtain multiple representations in a parameter-efficient backbone [35] where domain-specific FiLM [36] layers are inserted after each batch normalization layer. For addressing given few-shot tasks, Guo et al. [21] propose a greedy selection algorithm that iteratively searches for the best subset of features on all layers of all pre-trained models, and the selected features in the set are concatenated for training a linear classifier. SUR [27] proposes a feature selection procedure to linearly combine the domain-specific representations with different weights. URT [28] further trains a universal representation transformer layer to weigh the features. Different from [21,27,28] use multiple representations, URL [29] proposes distilling knowledge from the separate multi-domain networks into a single feature extractor.

Aside from directly using representations trained from one domain or multiple domains, some work has also looked into how to make effective few-shot task adaptations with limited data [19,20,37–40]. Concretely, TADAM [20] applies a task embedding network block, which takes the mean vector of few-shot features as input and produces element-wise scaling and shift vectors to adjust each batch normalization layer, thus making the feature extractor task-specific. *FN* [38] directly fine-tunes the scaling and shifting parameters of batch normalization on few-shot data to adapt the feature extractor. *ConFeSS* [39] proposes to learn a task-specific feature masking module that can produce refined features for fine-tuning a target classifier and the feature extractor. *Associative alignment* [19] first selects a set of task-relevant categories from source data and conducts feature alignment between the selected source data and target data for network adaptation. PDA [41] proposes a proxy-based domain adaptation scheme to optimize the pre-trained representation and a novel few-shot classifier simultaneously. Instead of adjusting the pre-trained network, some methods [37,40] choose to incrementally learn some parametric modules for adaptation to novel tasks and leave the pre-trained parameters frozen. For example, Implanting [37] adds and learns new convolutional filters within the existing CNN layers. TSA [40] attaches residual adapters to each module of a pre-trained model and optimizes them from scratch on the few-shot data. Unlike these methods that perform adaptation by leveraging

auxiliary parametric modules [20,37,39,40] or additional data [19], our method provides a more effective adaptation scheme that directly optimizes the pre-trained representations with the limited target data.

Except for the widely investigated few-shot learning for regular image recognition, recent studies have also focused on other tasks, such as scene recognition [42], multi-label classification [43] and multi-modal learning [44]. In this paper, we aim at a practical FSL setting, namely multi-source cross-domain few-shot learning. Different from most existing methods that focus on either representation learning or adaptation on the few-shot data, we address the problem by focusing on both aspects: how to design a good multi-source representation network and how to adapt the representations to address cross-domain FSL in a wide range of scenarios.

2.2. Domain Adaptation

Domain adaptation (DA) typically aims at transferring knowledge from a data-rich source domain to an unlabeled target domain. Most existing DA approaches intend to learn invariant feature representations across two domains by distribution alignment [45,46] or adversarial learning [47]. Besides the single-source DA, our method is more relevant to the multi-source DA [48,49], which also intends to leverage knowledge from multiple source domains. To learn domain-invariant feature representations, these methods typically align domains pairwise based on a domain-shared feature extractor, where the learning framework is also similar to ours. Concretely, Xu et al. [49] leverages multiple domain discriminators to reduce the domain shift by adversarial learning, while [48] matches moments of feature distributions across all pairs of source and target domains. However, the addressed task in this paper intrinsically differs in both single-source and multi-source DA, where the source and target domains have the same classes (or label space). In contrast, we tackle the problem of few-shot learning, where the classes in the source and target domains do not overlap.

2.3. Contrastive Learning

Our few-shot adaptation strategy is highly inspired by the self-supervised contrastive learning by imposing instance discrimination [50–53] and supervised contrastive learning [54]. All those methods aim to learn a good universal representation from a large-scale dataset, thus boosting transferability on a variety of computer vision tasks. A basic idea of these methods is to make contrasts between positive and negative pairs. For instance, NCE [50] proposes a non-parametric softmax classifier that is made up of instance features to achieve instance discrimination. MoCo [52] and SimCLR [53] typically construct different views from the same instance via a variety of data augmentations. SimCLR [53] learns the representation by minimizing the distance of the features from these views and maximizing the distance of the features from other instances. MoCo [52] minimizes contrastive loss based on a dynamic feature dictionary and a momentum encoder. Supervised contrastive learning [54] minimizes the distance of the features of the same category samples and maximizes the distance of the features from different categories. Unlike these methods that make use of contrastive learning for large-scale pre-training, we propose two contrastive objectives to impose both instance discrimination and class discrimination on the few-shot data for adapting pre-trained feature representations to be task-specific.

2.4. Multi-Task Learning

Multi-task learning aims to learn multiple related tasks simultaneously [35,55,56]. The main idea is to build a compact network that can represent all domains by sharing most model parameters except for minimal parameters for task specialization. Unlike multi-task learning, which aims to achieve optimal performance across multiple source tasks, transfer learning focuses on addressing a specific target task with insufficient training data using knowledge from a single or multiple source domain. In this paper, we seek

efficient multi-source representation learning in a multi-task setting in order to further support a broad range of downstream few-shot learning tasks.

3. Preliminary

3.1. Task Formulation

Few-shot image recognition aims to generalize basic knowledge to perform novel class categorization in previously unseen domains. It can be defined from an inductive transfer learning perspective, corresponding to two learning routines, i.e., meta-training and meta-testing stages. In the conventional few-shot setting, there is only one source dataset for training, and the deployed recognition scenario is also similar to the source domain, which is also regarded as *in-domain* few-shot learning. In contrast to the usual single-source setting, we look at how to use more abundant modal information from multiple source datasets sampled from different domains to support broad FSL tasks, especially for visual recognition tasks that go beyond natural images, such as remote sensing and medical images.

Formally, let us assume we have B source datasets $\mathcal{D} = \{\mathcal{D}_b\}_{b=1}^{B}$ in the meta-training stage, where each $\mathcal{D}_b = \{(\mathbf{x}, y)\} \subset \mathcal{X}_b \times \mathcal{Y}_b$ corresponds to a specific domain, and the $(\mathbf{x}, y)$ denotes an image sample and its associated class label. Based on deep neural networks, few-shot learning algorithms aim to extract general and transferable knowledge from large-scale data $\mathcal{D}$. In the meta-testing stage, the pre-trained model is adapted with a novel few-shot learning task which provides a small support set $\mathcal{S}$ sampled from target domain $\mathcal{D}_n = \{(\mathbf{x}, y)\} \subset \mathcal{X}_n \times \mathcal{Y}_n$. The configuration with the support set $\mathcal{S}$ containing N different classes with K samples each ($\mathcal{S} = \{S_i\}_{i=1}^{N}, |S_i| = K$) is referred to as the "N-way K-shot" recognition task. Particularly, all the source datasets $\mathcal{D}_b$ and target dataset $\mathcal{D}_n$ have no common class. After the pre-trained model has been adapted to the support set, a query set $\mathcal{Q}$ sampled from the unseen classes is used to evaluate the generalization performance.

3.2. Transfer Learning Baseline

We revisit a conventional transfer learning baseline, where a feature extractor F is first pre-trained on the source data and then frozen when adapting to the few-shot task. For the multi-source setting, we can simply merge the multiple source datasets into a joint dataset

$$\mathcal{D}_J = \mathcal{D}_1 \cup \mathcal{D}_2 \cup \cdots \mathcal{D}_b \cup \cdots \mathcal{D}_B \subseteq \mathcal{X}_J \times \mathcal{Y}_J \,. \tag{1}$$

Therefore, the representation learning can be conducted by one joint classification task. Associated with a classification layer C_{base}, the feature extractor can be trained in an end-to-end manner for recognizing all joint classes by minimizing the expected empirical risk,

$$F = \underset{F, C_{base}}{\arg\min} \underset{(\mathbf{x}, y_J) \sim \mathcal{D}_J}{\mathbb{E}} L(C_{base} \circ F(\mathbf{x}), y_J) \,, \tag{2}$$

where the L denotes a loss function (typically as cross-entropy) that measures the agreement between the true class label and the corresponding prediction from the classifier. Note, that the y_J denotes the class label in the current joint space $\mathcal{Y}_J$ instead of the space in the original source domain.

3.2.1. FT Baseline

Given a target few-shot task presented by $\mathcal{S}$, a simple fine-tuning (FT) baseline is freezing the pre-trained feature extractor and retraining a new classifier head C_{novel} with the features of the support set, i.e.,

$$C_{novel} = \underset{C_{novel}}{\arg\min} \underset{(\mathbf{x}, y) \sim \mathcal{S}}{\mathbb{E}} L(C_{novel} \circ F(\mathbf{x}), y) \,. \tag{3}$$

The new recognition model composed of $\{F, C_{novel}\}$ can be used for a target task. This baseline method has recently been proven to be effective when the pre-trained features can be transferable and reused in the target domain.

3.2.2. NNC Baseline

Another natural approach to directly performing unseen class categorization can resort to the k-nearest neighbor (KNN) with the pre-trained representation. Particularly, this method expects that the deeply learned features are discriminative and generalized enough to separate new classes, so the query (test) sample can be well classified by its nearest neighbors. Besides, a more generally used non-parametric method for multi-shot FSL is the nearest neighbor classifier (NNC) baseline [12,27], where the weights of the target classifier can be regarded as class prototypes [9]. Each prototype is computed using the averaged features of the corresponding support class as follows,

$$p_k = \frac{1}{|S_k|} \sum_{(\mathbf{x}, y=k) \in S_k} F(\mathbf{x}) \,. \tag{4}$$

For a query image $\mathbf{x}_q \in \mathcal{Q}$, the NNC assigns it to the label of the closest support class with a similarity metric $\mathrm{sim}(\cdot, \cdot)$ on the representation space,

$$\hat{y}_q = \underset{j \in \{1,\ldots,N\}}{\arg\max}\, \mathrm{sim}\big(F(\mathbf{x}_q), p_j\big) \,. \tag{5}$$

4. Approach

In this section, we elaborate on our approach to addressing multi-source cross-domain few-shot learning, which includes two learning stages: (1) learning multi-source representations and (2) adapting them to the few-shot task. Particularly, in the adaptation procedure, the objective is to adapt the pre-trained representations to be task-specific and discriminative enough for identifying novel classes.

4.1. Multi-Source Representation Learning

Given the multiple source datasets $\{\mathcal{D}_b\}_{b=1}^{B}$, we first present our framework for training multi-source representations, aiming to effectively extract the diverse semantic information. Typically, a simple way to obtain multi-domain representations is to train a separate feature extractor for each source domain [57]. However, when the number of domains is large, adapting and deploying a lot of models would be impractical. Besides, another baseline (presented in Section 3.2), training a single-task network with the merged source data, can be parameter-efficient but suppress feature diversity. What is worse, the potential interference across different domains may impede regular training [58].

To reduce the computational cost and model size, we achieve efficient domain-specific representations by a multi-head structure, where the multiple source datasets share a backbone network, with the assumption that low-level features are generalizable across different domains and tasks [5]. Concretely, the multi-head representations have multiple projection layers, each of which corresponds to a different domain and is used to map shared features into the space of that domain. The learning framework is depicted in Figure 2. Besides the original B domains, we also create a universal domain on all merged source data $\mathcal{D}_J$ presented in Section 3.2 and further define the number of feature representations as $D = B + 1$.

Inspired by the previous study [59], we instantiate each projection layer with a low-rank bilinear pooling (LBP) structure, since it has been proven to improve feature discrimination for the single-source FSL. Assuming the feature maps outputted by a shared CNN backbone are $f_\phi(\mathbf{x}) \in \mathbb{R}^{h \times w \times c}$, we add parallel low-rank bilinear (LBP) layers [60,61] at the end of the shared backbone $f_\phi(x)$ as can be seen in Figure 2. Denoting the D domain-specific LBP layers as $\{P_{\theta_i}\}_{i=1}^{D}$, we can obtain a feature representation set as $\{F_i(\cdot)\}_{i=1}^{D}$ for D domains,

$$F_i(\mathbf{x}) = P_{\theta_i} \circ f_\phi(\mathbf{x}) = \sum_{l=1}^{hw} P_{i,1}^T f_\phi(\mathbf{x})_l \odot P_{i,2}^T f_\phi(\mathbf{x})_l \,, \tag{6}$$

where $P_{i,1} \in \mathbb{R}^{c \times d}$ and $P_{i,2} \in \mathbb{R}^{c \times d}$ are two projection matrices for P_{θ_i} at the i-th domain, and subscript l represents the spatial index among $h * w$. As shown in Figure 3, the detailed architecture of LBP in our implementation consists of two parallel 1×1 convolutions with c channels followed by a Hadamard product and a global average pooling operation. The feature dimension can be manually set to d.

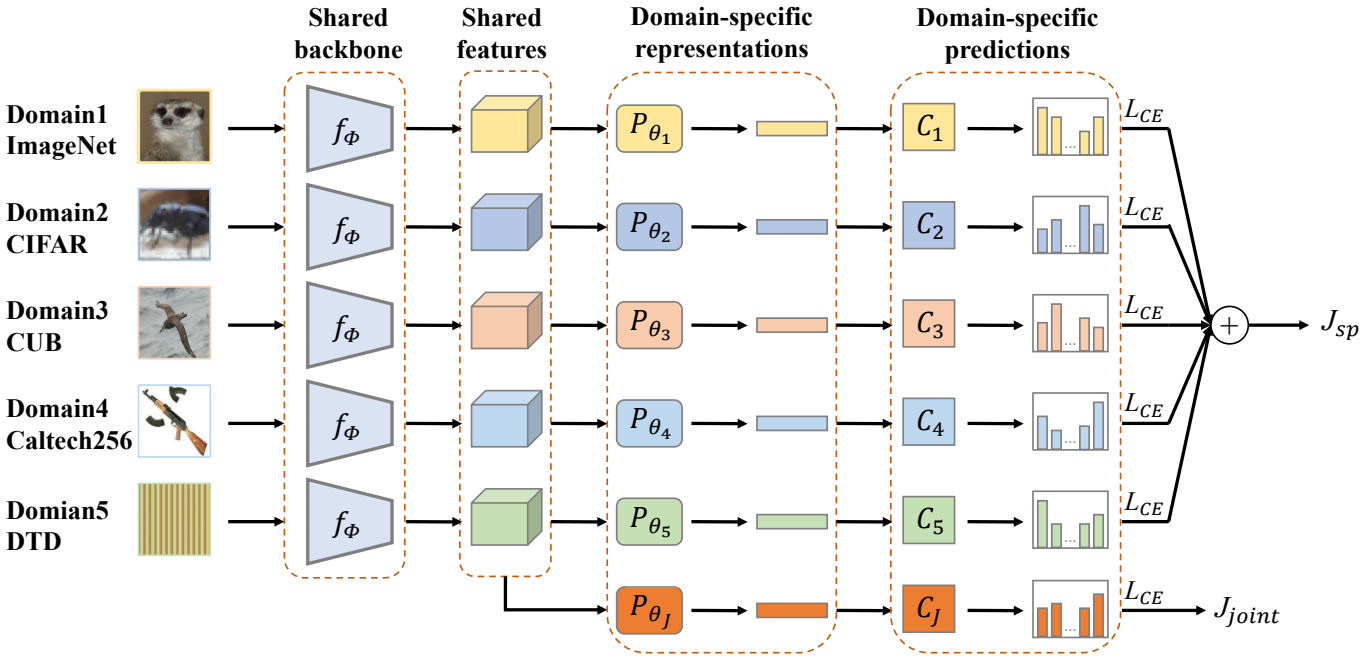


Figure 2. Multi-source representations learning from multiple source datasets. The structure of representations contains a shared CNN backbone f_ϕ and the multi-head projection layers $\{P_{\theta_i}\}_{i=1}^D$, which consists of a feature set $\{F_i(x)\}_{i=1}^D$. We train the representations by taking entropy loss on the multi-head classification tasks. The five input images correspond to the five source datasets of BSCD-FSL [21]. Best viewed in color.

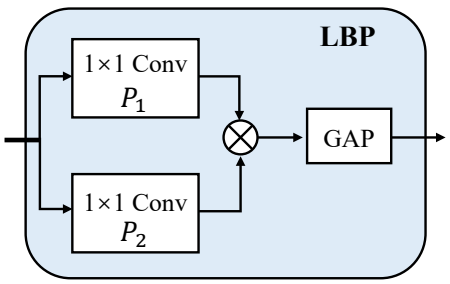


Figure 3. Structure of Low-rank bilinear pooling layer. This is used as the projection layer for achieving domain specialization. GAP denotes global average pooling.

We train the multi-source representations by using regular supervised training with in-domain classification, which is performed on each representation head. Concretely, the cosine classifiers [62,63] are used as the classification layers, denoted as $C = \{C_i(\cdot; W_i)\}_{i=1}^D$, where $W_i = [w_1, \ldots, w_{N_i}]$ are the d-dimensional classification weight vectors for the N_i classes in the i-th domain. The classifier $C_i(\cdot; W_i)$ produces the normalized classification score (probability) for the j-th class

$$C_i^j(F_i(\mathbf{x}); W_i) = \text{softmax}_j[\gamma \,\text{sim}(F(\mathbf{x}), w_i)]\,, \tag{7}$$

where the cosine similarity $\text{sim}(x, y) = \frac{x^T y}{\|x\|_2 \|y\|_2}$ is defined as the dot product between the two ℓ_2 normalized vectors, and γ is a regular associated scalar. In summary, the pre-training procedure minimizes the multi-domain classification losses. For clarity, we re-denote an image example in the joint dataset as $(\mathbf{x}, y_O, y_J, y_D) \sim \mathcal{D}_J$, where y_O, y_J, y_D are its original-domain class label, joint-domain class label, and domain index label, respectively. The end-to-end training objective in a multi-task setting is as follows,

$$J_{sp} = \sum_{b=1}^{B} \mathbb{1}_{[y_D=b]} * L_{CE}(C_b \circ F_b(\mathbf{x}), y_O)\,, \tag{8}$$

$$J_{joint} = L_{CE}(C_J \circ F_J(\mathbf{x}), y_J)\,, \tag{9}$$

$$F = \arg\min_{F,C} \mathop{\mathbb{E}}_{(\mathbf{x}, y_O, y_J, y_D) \sim \mathcal{D}_J} [J_{sp} + J_{joint}]\,, \tag{10}$$

where L_{CE} is the cross-entropy function, and $\mathbb{1} \in \{0, 1\}$ is a domain indicator function that returns 1 if its argument is true and 0 otherwise.

During network training with mini-batch SGD, the back-propagated gradients accumulated from the multiple tasks on the shared parameters may be too large to ensure proper end-by-end optimization. To stabilize the training process, we adopt a simple gradient scaling mechanism. To be specific, when the losses are backpropagated to the shared features, the cumulative gradients from the multi-head branches are averaged. In this way, the magnitude of gradients for domain-shared parameters (the CNN backbone) and domain-specific parameters (the projection and classification heads) can be balanced for proper end-to-end training.

In summary, our framework can be trained end-to-end and be built upon any CNN backbone, which is parameter-efficient and simple to implement. The joint training regime ensures that the shared low-level features are general and that multi-head projections are fully responsible for domain specialization. Thus, the produced representations can be universal enough to support further generalization to vastly different few-shot recognition tasks.

4.2. Adapting Representations on Few-Shot Data

After obtaining a set of feature representations $\{F_i(\cdot)\}_{i=1}^D$, we further conduct model adaptation, aiming to generalize the pre-trained representations to address the few-shot task, which only provides a small support set. To achieve this goal, we identify instance discrimination and class discrimination as two crucial factors for improving model generalization. Accordingly, we propose two contrastive learning objectives, which are performed on each domain-specific head. Different from the previous method [34], which uses contrastive learning on the source data to improve feature transferability in the first stage, our method conducts model adaptation by enhancing contrast across few-shot data, thus directly making the pre-trained features more specific and discriminative to the target task. As the adaptation procedure is conducted on each independent representation head $F(\cdot; \theta_i)$, we omit the domain index in the following notations for clarity. The adaptation procedure is depicted in Figure 4.

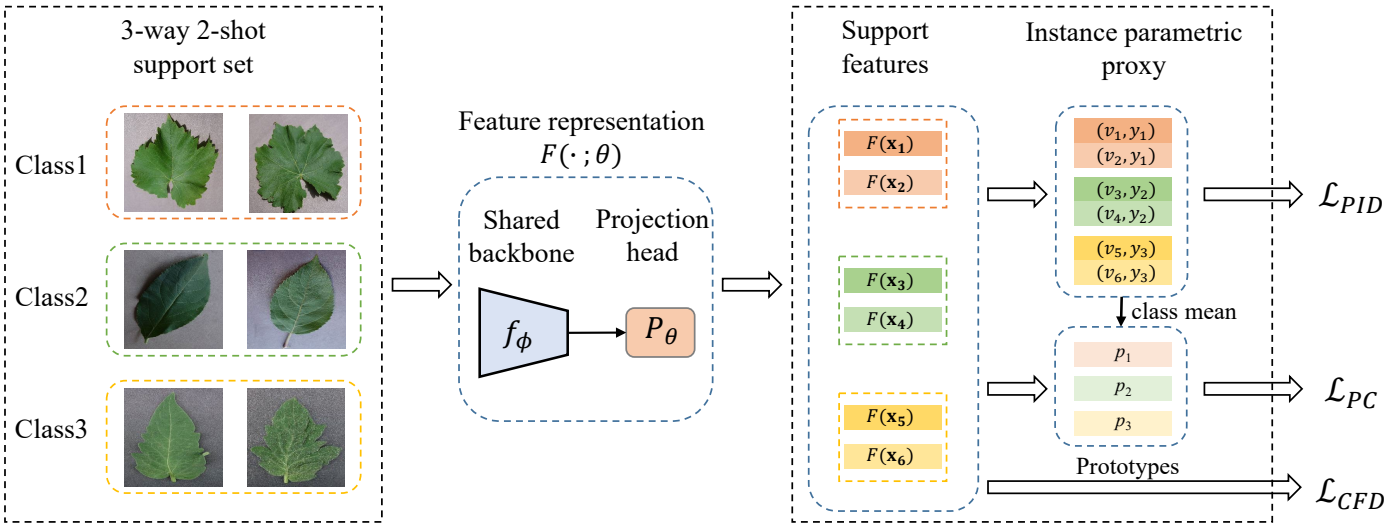


Figure 4. Adapting representations for recognizing previously unseen categories. The adaptation is performed on each representation head. Best viewed in color.

4.2.1. Parametric Instance Discrimination

Unlike most self-supervised contrastive losses, which use complex data augmentation to construct positive pairs from the same instance to achieve instance discrimination, we propose a parametric module, namely instance parametric proxy (IPP). It is functionally similar to a memory bank storing instance features for instance classification [50,52], but different in that our IPP is learnable and updated by gradient descent. For an N-way K-shot task that provides a support set $\mathcal{S} = \{S_i\}_{i=1}^{N}, |S_i| = K$, we denote the weights of the IPP as $V = \{v_i\}_{i=1}^{N*K}, v_i \in \mathbb{R}^d$, each of which corresponds to a support instance and can be initialized by original support features. For each iteration of model adaptation, let $i \in I \equiv \{1, \ldots, N * K\}$ be the index of an arbitrary transformed sample from the original support image, and $\mathcal{A}(i)$ be the negative index set of the sample $\mathbf{x}_i$. Then, we perform contrastive learning by enforcing each instance $\mathbf{x}_i$ to be close to its proxy v_i and far from its negative samples indexed from $\mathcal{A}(i)$ in the feature space. Our parametric instance discrimination (PID) loss modified from info-NCE [51] is as follows:

$$\mathcal{L}_{PID}(\mathcal{S}; \phi, \theta, V) = \frac{1}{|I|} \sum_{i \in I} -\log \hat{P}_{i,p}\,, \tag{11}$$

$$\hat{P}_{i,p} = \frac{\mathrm{E}^{(F(\mathbf{x}_i), v_i)}}{\mathrm{E}^{(F(\mathbf{x}_i), v_i)} + \sum\limits_{a \in \mathcal{A}(i)} [\mathrm{E}^{(F(\mathbf{x}_i), v_a)} + \mathrm{E}^{(F(\mathbf{x}_i), F(\mathbf{x}_a))}]}\,, \tag{12}$$

where $\mathrm{E}^{(w_1, w_2)} = \exp(\mathrm{sim}(w_1, w_2)/\tau)$ and τ is a regular temperature parameter. Unlike unsupervised contrastive learning [52,53] makes an anchor instance discriminate from all other instances; the negative index set in Equation (12) is defined as $\mathcal{A}(i) \equiv \{a \in I : y_a \neq y_i\}$, which means the negative pairs between same-class instances would be filtered by the category labels. As a result, $|\mathcal{A}(i)| = (N-1) * K$ for the N-shot K-shot task specifically. Thus, this supervised objective is more effective in reducing instance variations, benefiting from avoiding negative contrast between same-class features. During adaptation, this contrastive loss is jointly minimized with respect to the IPP and the parameters of feature representation by SGD.

4.2.2. Class Feature Discrimination

While instance discrimination loss can enhance instance invariance for improving generalization, it only loosely ensures intra-class compactness [54], which is a key capability to cluster same-class features. To make the representations more discriminative to the target task, we enforce intra-class feature invariance while also keeping the separation of between-class features. Given the arbitrary transformed support samples indexed by $i \in I \equiv \{1...N * K\}$, let $\mathcal{A}(i)$ and $\mathcal{P}(i)$ for sample $\mathbf{x}_i$ be the negative and positive index sets, respectively. Here, $\mathcal{P}(i) \equiv \{p \in I : y_p = y_i\} \setminus \{i\}$ and $|\mathcal{P}(i)| = K - 1$ for the K-shot task specifically. Then our class feature discrimination (CFD) targets minimizing the supervised contrastive loss as follows,

$$\mathcal{L}_{CFD}(\mathcal{S}; \phi, \theta) = \frac{1}{|I|} \sum_{i \in I} \frac{1}{|\mathcal{P}(i)|} \sum_{p \in \mathcal{P}(i)} -\log \hat{P}_{i,c}\,, \tag{13}$$

$$\hat{P}_{i,c} = \frac{\mathrm{E}^{(F(\mathbf{x}_i), F(\mathbf{x}_p))}}{\mathrm{E}^{(F(\mathbf{x}_i), F(\mathbf{x}_p))} + \sum\limits_{a \in \mathcal{A}(i)} \mathrm{E}^{(F(\mathbf{x}_i), F(\mathbf{x}_a))}}\,. \tag{14}$$

With the complementary supervision of Equations (11) and (13), not only the inter-class feature differences can be enlarged but also the intra-instance and intra-class feature variations can be reduced.

4.2.3. Prototypical Classification

The two contrastive objectives can enhance model discrimination at the feature level, thus improving accuracy for the direct NNC baseline. However, previous literature findings [14,64] also indicate that training linear classifiers on the frozen feature extractor can outperform NCC, as they can learn better class boundaries by exploiting overall support examples instead of only calculating class centers. A natural approach is to build a linear classifier from scratch as presented *FT baseline* in Section 3.2. Here, we propose to implicitly conduct classification by repurposing the IPP without rebuilding a parametric layer. It is also beneficial to avoid over-parameterization in the low-data regime. In each iteration, we first calculate the class prototypes by averaging the instance proxies belonging to the same class:

$$p_k = \frac{1}{K} \sum_{(v_i \sim V, y_i = k)} v_i,\ k = 1, \ldots, N\,. \tag{15}$$

For a support sample $\mathbf{x}_i$, the posterior probability $\hat{P}_s$ belonging to the support class k is as follows

$$\hat{P}_s(y = k | \mathbf{x}_i) = \frac{\exp(\mathrm{sim}(F(\mathbf{x}_i), p_k))}{\sum_j \exp\big(\mathrm{sim}\big(F(\mathbf{x}_i), p_j\big)\big)}\,. \tag{16}$$

The prototypical classification loss based on cross-entropy between the predictions and the support labels is as follows

$$\mathcal{L}_{PC}(\mathcal{S}; \phi, \theta, V) = \sum_{(\mathbf{x}_i, y_i) \in \mathcal{S}} -\log \hat{P}_s(y = y_i | \mathbf{x}_i)\,. \tag{17}$$

This regularization can encourage the model to learn more comprehensive features by enforcing accurate prediction with natural cross-entropy. It turns out to be particularly effective and leads to significant performance boosts, which can be attributed to improvements in the quality of the adapted features and more representative class prototypes induced by IPP.

4.2.4. Implementation of Total Adaptation Loss

Finally, the few-shot adaptation is conducted by minimizing the three combined losses on the support set:

$$\mathcal{L}_{total}(\mathcal{S}; \phi, \theta, V) = \mathcal{L}_{PID} + \lambda_1 * \mathcal{L}_{CFD} + \lambda_2 * \mathcal{L}_{PC}, \tag{18}$$

where the λ_1 and λ_2 are two regular trade-off parameters.

We conduct two adaptation strategies: (1) *LAMR*: adapting the projection layers and leaving the backbone frozen. This adaptation is independently performed on each representation head with the shared features extracted, which typically allows for rapid adaptation. (2) *LAMR++*: adapting both the projection layers and the shared backbone.

4.2.5. Query Prediction

With the proposed adaptation, we can transfer the pre-trained representations to task-specific ones, denoted as $\{F(\cdot; \hat{\theta}_i)\}_{i=1}^{D}$. We can build the nearest neighbor classifier (NNC) [9] by each adapted IPP. For the i-th domain, the induced prototypes of NCC computed by Equation (15) are denoted as $\{\hat{p}_j^i\}_{j=1}^{N}$. For a query image $\mathbf{x}_q \in \mathcal{Q}$, the similarity to the class j on the i-th representation is computed as $\text{sim}(F(\mathbf{x}_q; \hat{\theta}_i), \hat{p}_j^i)$. The final prediction is based on the aggregation of class similarity across all the representation heads, as follows:

$$\hat{y}_q = \underset{j \in \{1,\ldots,N\}}{\arg\max} \sum_{i=1}^{D} \text{sim}(F(\mathbf{x}_q; \hat{\theta}_i), \hat{p}_j^i) \tag{19}$$

4.3. Extension to Single-Source FSL

We can make our multi-source framework applicable to the single-source FSL by dividing the source dataset into sub-domains, each of which contains some unique classes. We propose the following two splitting methods.

1. Random splitting. The original classes are equally randomly split into sub-datasets.
2. Clustering splitting. One natural class splitting choice would be K-means clustering on class prototypes computed over image features, with a representation pre-trained on the full classes. However, K-means may result in unbalanced partitions. Inspired by the previous method [65], we iteratively split each current dataset in half along the principal component computed over the class prototypes. For splitting N iterations, the original dataset can be divided into 2^N subsets, each of which can be regarded as a distinct domain and is composed of classes that are closer to each other.

With the split sub-domains on a fixed training data budget, our framework can indeed encourage more diverse representations, which can be further adapted to the FSL task and produce an ensemble of NNC classifiers. However, it typically worsens the performance of the individual classifier on average but makes the ensemble prediction more accurate. This modification turns out to be particularly effective when the number of partitions is appropriate, as illustrated in the experimental section. For the extreme case where there is only one class per domain, the representation cannot be trained without supervision. Therefore, we can expect that there may exist an optimal number of partitions for a dataset. We can choose the hyperparameter based on the performance of the validation set.

5. Experiments and Results

5.1. Benchmark Datasets

5.1.1. Broader Study of Cross-Domain Few-Shot Learning (BSCD-FSL)

We mainly evaluate our approach on the recently proposed cross-domain benchmark BSCD-FSL [21], which provides few-shot evaluation protocols in both single- and multi-source settings. Figure 5 shows examples of source and target images. For the multi-source setting, training datasets contain *mini*-ImageNet [8], CUB [66], CIFAR100 [67], DTD [68], and Caltech256 [69]. All the source domains belong to colored natural images. For the

single-source setting, only *mini*-ImageNet is used. The testing domain covers a spectrum of image types, including CropDiseases [70], EuroSAT [23], ISIC2018 [24,25], and ChestX [26] datasets. Concretely, they are images of plant diseases, remote sensing images, dermoscopy images of skin lesions, and chest X-ray images, each of which corresponds to a different level of similarity to natural images. Compared to previous benchmarks [13,22], this provides more diverse specialized recognition scenarios for evaluating cross-domain FSL.

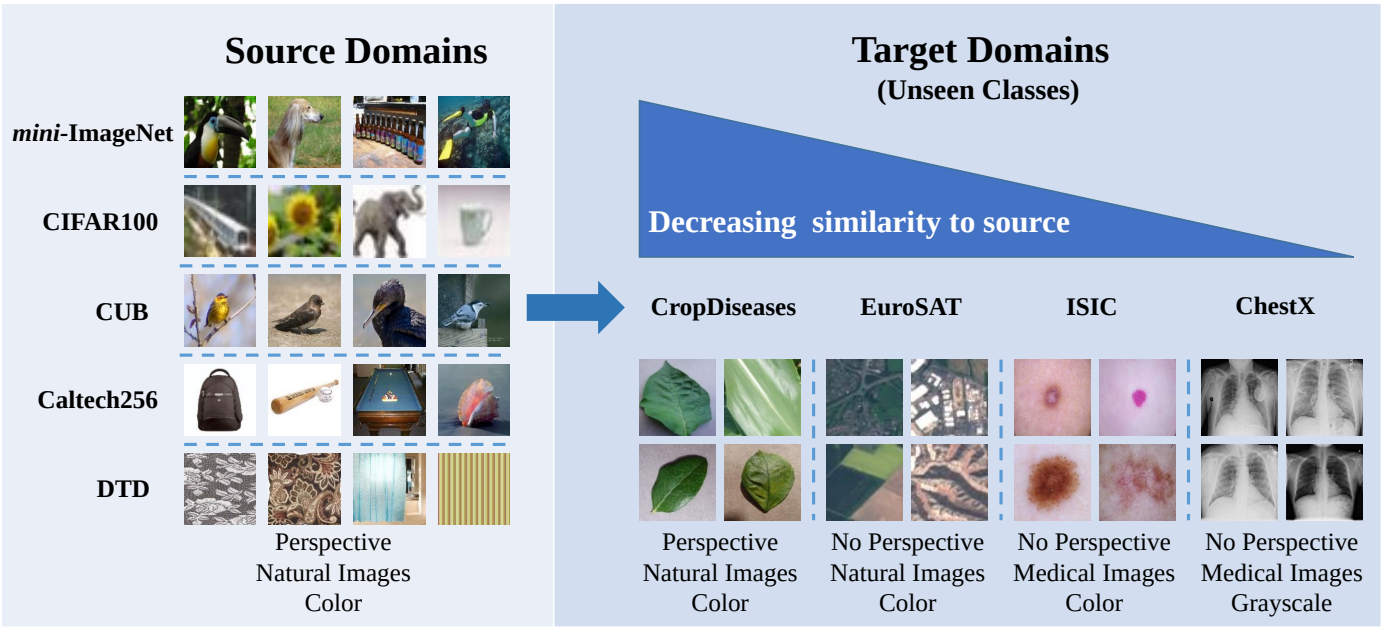


Figure 5. Image examples of datasets in the Broader Study of Cross-Domain Few-Shot Learning (BSCD-FSL) benchmark. Five training datasets are *mini*-ImageNet, CUB, CIFAR100, DTD, and Caltech256. Four testing domains are CropDiseases, EuroSAT, ISIC2018, and ChestX.

5.1.2. *Mini*-Imagenet

For conventional (in-domain) few-shot learning, we evaluate the most commonly used *mini*-ImageNet [8] dataset, which is derived from the ImageNet dataset [3] and consists of 60,000 color natural images of size 84×84 that belong to 100 classes, each with 600 examples. The *mini*-ImageNet was first proposed in [10]. We use the common follow-up setting [8] where the dataset is divided into 64 base classes, 16 validation classes, and 20 novel classes. To make this dataset applicable to our framework, the proposed splitting methods in Section 4.3 are performed on the 64 base classes, and the optimal hyperparameters are selected by the validation set.

5.2. Implementation Details

5.2.1. Network Architecture

We use ResNet12 [20,64], a derivative of residual networks [2] particularly designed for few-shot learning, as the feature extraction backbone $f_\phi(\cdot)$ to produce the shared features in all experiments. The detailed structure of ResNet12 is shown in Figure 6. ResNet12 has four residual blocks, and each block is made up of three convolutional layers and one 2×2 max-pooling layer with stride 2. Each convolutional layer has a 3×3 kernel, followed by batch normalization and leaky ReLU of 0.1. The four blocks output feature maps with 64/160/320/640 channels, respectively. The number of parameters of ResNet12 is approximately 12.4M. For the $3 \times 84 \times 84$ input image size, the output feature maps have a size of $640 \times 5 \times 5$. The low-rank bilinear pooling (LBP) layer utilizes two parallel 1×1 convolutional layers. We set its feature dimension d as 640 equal to the output channels of the ResNet12 backbone. The number of parameters of the LBP is approximately 0.8 M.

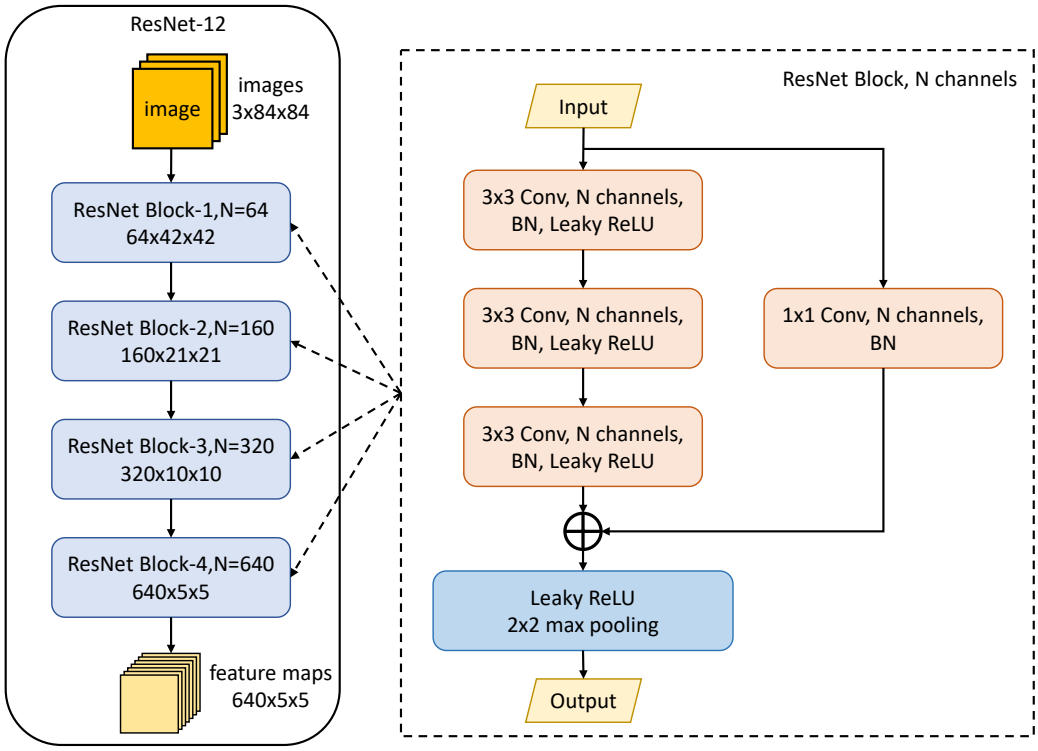


Figure 6. The detailed structure of ResNet12. It contains four residual blocks, each of which is made up of three {3 × 3 convolution (Conv), batch normalization (BN), Leaky ReLU of 0.1} and one 2 × 2 max-pooling layer.

5.2.2. Training Details

Our codebase is developed on the few-shot learning framework with Pytorch in [21]. In the meta-training stage, we use the SGD optimizer with the Nesterov momentum 0.9 and a weight decay of $1 \times e^{-4}$ is applied to all the model parameters. We train 140 epochs totally on both single- and multi-source learning, with the learning rate initialized by 0.1 and dropped to 0.01 at the 100 epoch similar to [12]. Conventional data augmentations, including random resize and crop, horizontal flip, and color jittering are applied to the source training images. In the meta-testing stage, we conduct model adaptation on the few-shot data. For the method *LAMR*, only the domain-specific layers are fine-tuned. For the method *LAMR++*, both the pre-trained backbone and the domain-specific layers are fine-tuned. Concretely, we use an SDG optimizer with 100 epochs fine-tuned on few-shot data (support set). The trade-off parameters λ_1 and λ_2 are simply set to 1. The metric scalar γ (or temperature parameter τ) in the cosine similarity is set to 20 (or 0.05) in all equations.

5.2.3. Evaluation Protocol

For the BSCD-FSL benchmark, we evaluate 5-way few-shot performance varying the shot in {5, 20, 50} over the 600 tasks following the previous evaluation protocol [21]. For the *mini*-ImageNet benchmark, we evaluate 5-way 1-shot and 5-shot generalization performance over 2000 tasks on the novel set as in [18,41]. Each few-shot task contains 15 queries per class. We report the average accuracy with a corresponding 95% confidence interval over all tasks for all experiments. In particular, we apply consistent sampling to make a fair comparison rigorously, where the sampling of testing few-shot tasks follows a deterministic order by *numpy* with a fixed seed. It makes our ablation studies and comparisons more convincing.

5.3. Results on Multi-Source FSL

We present the results of our method in the multi-source FSL setting, where five semantically different datasets can be used for pre-training.

The following multi-domain learning methods are compared:

- *Union-CC* [62]: A baseline method trains a single feature extractor on the union of all training data with the cosine classifier and tests it with the NNC classifier.
- *Ensemble*: A baseline method trains separate feature extractors on each dataset and tests with the average prediction of the NCC classifiers built on them.
- *All-EMDs* [21]: A method concatenates the feature vectors of all layers of all the separate feature extractors for training a linear classifier.
- *IMS-f* [21]: A greedy selection method iteratively searches for the best subset of features on all layers of all the separate feature extractors for a given few-shot task. Then, the feature vectors in the set are concatenated for training a linear classifier.
- *FiLM-pf* [35]: Multiple representations are trained on a parameter-efficient backbone, in which parallel domain-specific FiLM [36] layers are inserted after each batch normalization layer. The multiple representations are tested with the average prediction of NCC classifiers built on them.
- *SUR* [27]: A feature selection method performs a linear combination of domain-specific representations on the *FiLM-pf*.
- *URL* [29]: A single feature extractor is distilled from the separate multi-domain networks and tested with the NNC classifier.
- *URL+Ad* [29]: An adaptation method attaches a *pre-classifier feature mapping* (a linear layer) to the *URL* and optimizes it with the few-shot data.
- *TSA* [40]: An adaptation method attaches residual adapters to each convolution layer of a pre-trained model (here is *URL* [29]) and a *pre-classifier feature mapping* on the pre-trained model and optimizes them from scratch with the few-shot data.

Table 1 reports the detailed results on the four target datasets. Figure 7 summarizes the result comparison across different methods according to the average accuracy across all shot levels and datasets in the benchmark. We can observe that the proposed *LAMR* demonstrates a clear promise to set a new state of the art in all experimental settings, as it can consistently precede both previous methods and the baseline methods, as shown in Table 1. Concretely, *Ensemble*, *All-EBDs*, and *IMS-f* use the multi-domain features built on the fully separated feature extractors, which is inefficient and impractical when deployed to target domains. In addition, *FiLM-pf* and *SUR* built on a parameter-efficient backbone are also computationally expensive, as they still require multiple forward passes to obtain the multi-domain features. In contrast to these methods, our multi-head network is both parameter-efficient and computation-efficient. Instead of using an ensemble of multiple feature representations, *URL* learns a single network by knowledge distillation from the ensemble of separate multi-domain networks. The distilled single network shows better generalization ability compared to the *Ensemble* and *Union-CC* but still underperforms compared to other methods if directly using its features. However, when making further adaptations to the *URL*, results can be significantly improved. Concretely, *URL+Ad* simply employs a linear layer on the top of the *URL* to make feature adaptation, which results in an average improvement of 2.7%. *TSA* conducts a deeper adaptation, which can provide an average improvement of 3.9% over the *URL*. Besides, our proposed adaptation method is a combination of fine-tuning losses, which is also orthogonal to methods like *TSA* that make network adaptation by incrementally learning some new parametric modules. Last, our *LAMR*, which only fine-tunes the projection layers, consistently performs better than the *TAS* in all settings shown in Table 1. With the backbone being further adapted, our method *LAMR++* can achieve consistent performance gains. Particularly on the ISIC dataset, our *LAMR++* can produce improvements of 3.7%, 3.2%, and 3.4% over our shallower adaptation method (*LAMR*) in {5/20/50}-shot settings, respectively. It is generally considered important to adapt both shallow and deep layers in a neural network for successfully addressing cross-domain few-shot learning.

Table 1. The results of multi-source few-shot learning on the BSCD-FSL benchmark. Best results are marked in bold.

Methods	ChestX			ISIC		
	5-Way 5-Shot	**5-Way 20-Shot**	**5-Way 50-Shot**	**5-Way 5-Shot**	**5-Way 20-Shot**	**5-Way 50-Shot**
Union-CC [62]	26.08 ± 0.41	31.14 ± 0.43	33.54 ± 0.45	43.35 ± 0.55	51.71 ± 0.58	54.34 ± 0.53
Ensemble	26.45 ± 0.44	30.81 ± 0.45	33.47 ± 0.46	44.49 ± 0.57	52.49 ± 0.56	55.06 ± 0.54
All-EBDs [21]	26.74 ± 0.42	32.77 ± 0.47	38.07 ± 0.50	46.86 ± 0.60	58.57 ± 0.59	66.04 ± 0.56
IMS-f [21]	25.50 ± 0.45	31.49 ± 0.47	36.40 ± 0.50	45.84 ± 0.62	61.50 ± 0.58	68.64 ± 0.53
FiLM-pf [35]	26.79 ± 0.45	30.91 ± 0.45	33.80 ± 0.47	47.06 ± 0.56	55.43 ± 0.56	57.73 ± 0.53
SUR [27]	26.81 ± 0.46	30.98 ± 0.45	33.85 ± 0.46	47.37 ± 0.56	55.59 ± 0.59	57.92 ± 0.53
URL [29]	26.49 ± 0.45	30.40 ± 0.44	33.75 ± 0.46	46.00 ± 0.58	53.87 ± 0.58	56.32 ± 0.54
URL+Ad [29]	26.68 ± 0.44	31.41 ± 0.44	36.41 ± 0.45	48.10 ± 0.60	58.84 ± 0.63	64.16 ± 0.58
TSA [40]	27.04 ± 0.43	33.31 ± 0.47	37.15 ± 0.48	49.40 ± 0.61	62.34 ± 0.60	67.73 ± 0.56
LAMR	27.37 ± 0.41	34.16 ± 0.48	39.21 ± 0.51	52.58 ± 0.61	65.33 ± 0.55	70.52 ± 0.53
LAMR++	**28.38 ± 0.45**	**36.77 ± 0.50**	**42.22 ± 0.54**	**56.26 ± 0.66**	**68.52 ± 0.55**	**73.89 ± 0.52**
Methods	**EuroSAT**			**CropDiseases**		
	5-Way 5-Shot	**5-Way 20-Shot**	**5-Way 50-Shot**	**5-Way 5-Shot**	**5-Way 20-Shot**	**5-Way 50-Shot**
Union-CC [62]	81.01 ± 0.56	86.05 ± 0.48	87.30 ± 0.41	90.22 ± 0.54	93.97 ± 0.36	95.09 ± 0.32
Ensemble	84.03 ± 0.58	88.10 ± 0.49	88.44 ± 0.48	91.89 ± 0.51	95.04 ± 0.37	96.06 ± 0.30
All-EBDs [21]	81.29 ± 0.62	89.90 ± 0.41	92.76 ± 0.34	90.82 ± 0.48	96.64 ± 0.25	98.14 ± 0.18
IMS-f [21]	83.56 ± 0.59	91.22 ± 0.38	93.85 ± 0.30	90.66 ± 0.48	97.18 ± 0.24	98.43 ± 0.16
FiLM-pf [35]	83.93 ± 0.58	87.82 ± 0.51	87.94 ± 0.48	93.73 ± 0.46	96.18 ± 0.32	97.26 ± 0.25
SUR [27]	84.35 ± 0.59	88.32 ± 0.50	88.42 ± 0.49	93.72 ± 0.46	96.16 ± 0.33	97.26 ± 0.25
URL [29]	83.74 ± 0.58	88.52 ± 0.48	89.13 ± 0.45	92.13 ± 0.50	95.18 ± 0.36	96.21 ± 0.27
URL+Ad [29]	84.57 ± 0.55	91.66 ± 0.36	93.66 ± 0.31	93.12 ± 0.44	97.23 ± 0.24	98.51 ± 0.15
TSA [40]	85.10 ± 0.55	92.25 ± 0.34	94.24 ± 0.29	93.53 ± 0.44	97.58 ± 0.22	98.81 ± 0.13
LAMR	86.92 ± 0.47	93.65 ± 0.29	95.42 ± 0.23	94.61 ± 0.39	98.26 ± 0.18	99.12 ± 0.11
LAMR++	**87.38 ± 0.47**	**94.40 ± 0.26**	**96.31 ± 0.21**	**94.84 ± 0.39**	**98.57 ± 0.16**	**99.30 ± 0.10**

Overall, we can clearly observe that our methods (*LAMR* & *LAMR++*) have absolute superiority over other methods. Concretely, *LAMR++* achieves an average classification accuracy of 73.07% across all datasets and shot levels, which outperforms the *TSA* by 3.2%.

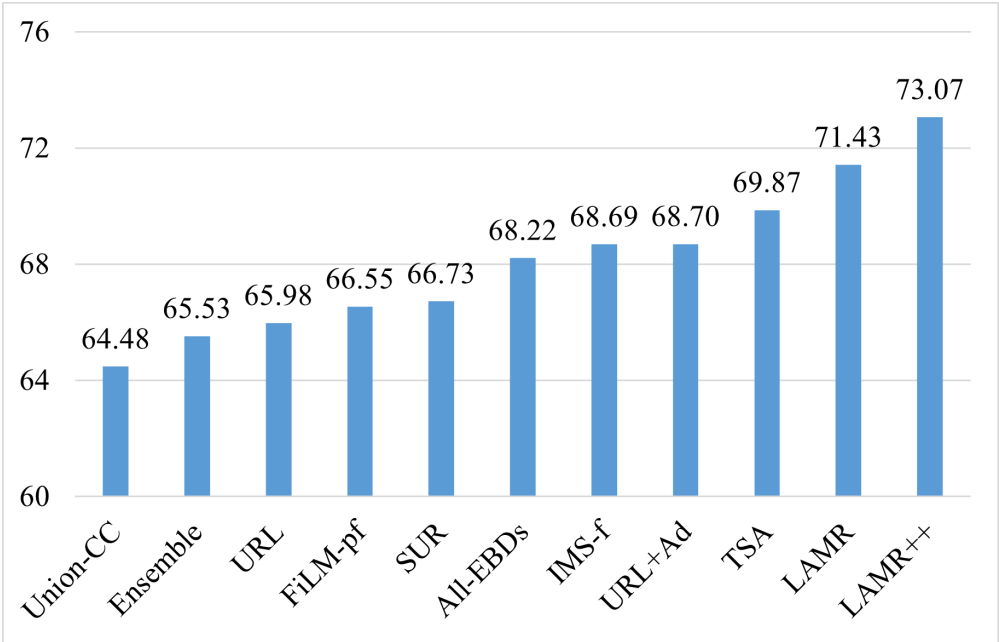
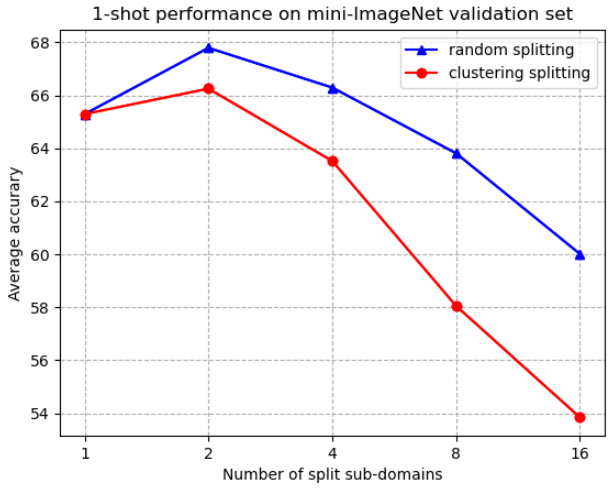


Figure 7. Overall performance comparison across different methods in the multi-source setting of BSCD-FSL benchmark.

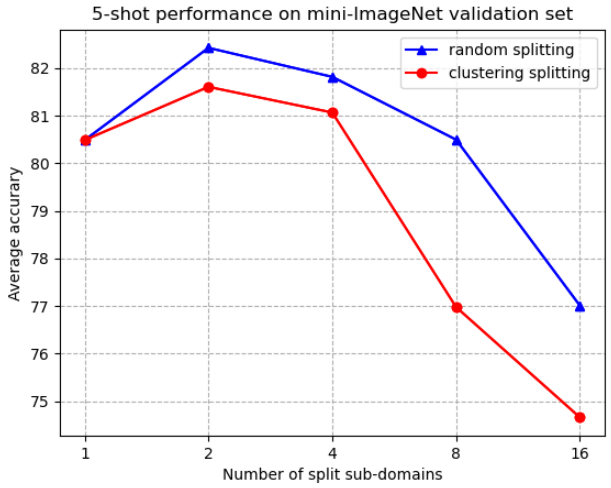
5.4. Results on Single-Source FSL

5.4.1. Validating Splitting Strategy for Single-Source FSL

We first investigate the effects of different splitting methods and splitting numbers for single-source FSL. As previously demonstrated in Section 4.3, there may exist a trade-off partition number for achieving the optimal generalization performance. The evaluation is conducted on the validation set of *mini*-ImageNet, with consistently sampled 200 tasks. We evaluate the two proposed splitting methods, random splitting and clustering splitting, with the splitting number varying in $\{1, 2, 4, 8, 16\}$. For the random splitting method, we perform three trials based on different random split classes and report the mean of the three runs. It is worth noting that the results across different trials have a low variance, as the variation between each two random trials is within 1%. The plots of five-way one-shot and five-shot validation accuracy are shown in Figure 8. The one splitting (or root point) refers to the strong FSL baseline, namely CC [62,63], trained with the original dataset without splitting.



(**a**) 5-way 1-shot validation accuracy on *mini*-ImageNet



(**b**) 5-way 5-shot validation accuracy on *mini*-ImageNet

Figure 8. Validation accuracy varying splitting strategies and numbers of partitioned sub-domains on *mini*-ImageNet. (**a**) Five-way one-shot validation performance (**b**) 5-way 5-shot validation performance. We can observe that the best performance is achieved at 2 splitting for both 1-shot and 5-shot settings, which is also a trade-off between the number of split domains and classes (or data) per domain for a fixed data budget on the *mini*-ImageNet benchmark.

First, it is of interest that the random splitting method achieves better performance than the clustering splitting method. A possible explanation for this phenomenon is that randomly split sub-domains include more heterogeneous classes, which can yield more discriminative representations and better average performance in the ensemble. Second, we can observe that the best number of splitting for both methods on *mini*-ImageNet is 2. It also indicates our multi-source framework can improve single-source FSL with a fixed data budget. Third, we can find that when the split sub-domains (greater than 4) become too large, the accuracy is seriously decreased, since there are too many limited classes and data to enable meaningful representation learning in each sub-domain.

5.4.2. Results on *Mini*-ImageNet

We further evaluate our LAMR trained on the "fake multi-domain" which is partitioned by the optimal splitting strategy validated in Figure 8. We report the results on the *mini*-ImageNet test set and compare them with the prior methods that focus on learning or adapting a good representation in Table 2. We make comparisons as follows. (1) First, we compare our method with the approaches [12–14,62,71] that directly rely on a good pre-trained representation learning. They all perform few-shot classification on the frozen

representation by building a target classifier with *NNC* [12,62,71] or *FT* [13,14] baselines, but differ in the way the feature extractor is learned. Concretely, the *CC* [62] is our baseline model, whose deep representation is trained on a cosine classifier. *Neg-Cosine* [71] enhances *CC* representation by employing a negative margin into the softmax loss. Other methods [12–14] would rather use natural linear classifiers to minimize the cross-entropy loss for obtaining the representation, and the *Meta-Baseline* [12] further improves the pre-trained feature extractor followed by a meta-training stage. All the methods mentioned above are competitive with meta-learning-based methods [9,64], and beneficial from a good embedding. However, our *LAMR* shows significant superiority over the *CC*-based methods [62,71] as well as other methods [12–14]. For example, *LAMR* can outperform *Embed-Distill* [14] by 1.9% and 1.8% in 5-shot and 1-shot settings, respectively. Besides, our method is also orthogonal with those methods [9,12,13,62,71], as their learning algorithms can also be used for pre-training our multi-head representation framework.

Table 2. Comparison to previous methods on *mini*-ImageNet. Our LAMR is trained following the optimal splitting strategy selected by the validation set, which can fairly compare with other methods under a same training data budget. All methods use ResNets as the feature backbone. The best result in each setting is marked in bold.

Type	Method	Backbone	5-Way 1-Shot	5-Way 5-Shot
w/o Adapt	ProtoNet [9] by [64]	ResNet12	59.25 ± 0.64	75.60 ± 0.48
	MetaOptNet [64]	ResNet12	62.64 ± 0.62	78.63 ± 0.46
	CC [62] by [37]	ResNet12	58.61 ± 0.18	76.40 ± 0.13
	baseline [13]	ResNet18	51.75 ± 0.80	74.27 ± 0.63
	Neg-Cosine [71]	ResNet12	63.85 ± 0.81	81.57 ± 0.56
	Embed-Distill [14]	ResNet12	64.82 ± 0.60	82.14 ± 0.43
	Meta-Baseline [12]	ResNet12	63.17 ± 0.23	79.26 ± 0.17
	Robust20 [57]	ResNet18	63.95 ± 0.42	81.59 ± 0.42
w/ Adapt	TADAM [20]	ResNet12	58.50 ± 0.30	76.70 ± 0.30
	Centroid-Align [19]	ResNet18	59.88 ± 0.67	80.35 ± 0.73
	Implant [37]	ResNet12	62.53 ± 0.19	79.77 ± 0.19
	DC+SUR [27]	ResNet12	63.13 ± 0.63	80.04 ± 0.41
	Free-Lunch [72]	ResNet12	64.73 ± 0.44	81.15 ± 0.42
	H-OT [73]	ResNet12	65.63 ± 0.32	82.87 ± 0.43
	LAMR (ours)	ResNet12	65.73 ± 0.43	83.37 ± 0.29
	LAMR++ (ours)	ResNet12	**65.90 ± 0.43**	**83.84 ± 0.29**

(2) Second, we further compare our *LAMR* with the methods [19,20,27,37,72,73] that perform feature adaptation when employed to target few-shot tasks. *TADAM* [20] employs a task embedding network (TEN) block that generates scaling and shift vectors for each batch normalization layer, adapting the network to be task-specific. However, learning an accurate auxiliary network may be a challenging task, especially when target data are limited and the domain shift is significant. *Centroid-Align* [19] first selects a set of task-relevant categories from source data and conducts feature alignment between the selected source data and target few-shot data for network adaptation. *Free-Lunch* [72] proposes to calibrate the distribution of the novel samples using the statistics of selected base classes that are considered task-relevant. *H-OT* [73] further develops a novel hierarchical optimal transport framework to achieve adaptive distribution calibration. Unlike methods [19,72,73] that perform adaptation by leveraging the base data [19] or their statistics [72,73], our method provides a more effective adaptation scheme that directly optimizes the pre-trained representations with the limited target data without re-accessing the source data. Besides, our methods also achieve on par with or better performance than them. For example, our *LAMR*++ outperforms *H-OT* by 0.27% and 0.97% in 1-shot and 5-shot settings, respectively. Instead of adjusting the pre-trained parameters, *Implant* [37] adds and learns new convolutional filters upon the frozen CNN layers. Our *LAMR*++ is also orthogonal

to it and performs significantly better than it by 3.4% and 4.1% in 1-shot and 5-shot settings, respectively.

(3) Third, we compare *LAMR* with an ensemble-based method, Robust-20 [57], which trains an ensemble of 20 ResNets promoted by cooperation and diversity regularization. Our approach is more efficient for building an ensemble of multiple representations and also significantly outperforms it, with notable absolute accuracy improvements of 2.0% and 2.3% for 1-shot and 5-shot settings, respectively.

It is observed that adapting the backbone (by LAMR++) only slightly helps few-shot transfer for this benchmark dataset, whereas the improvement is more pronounced for cross-domain few-shot learning. It may also indicate that deeper adaptation is more necessary when the domain distribution shift increases.

5.4.3. Results on BSCD-FSL

We further report the results of single-source cross-domain FSL on the four specific domains of BSCD-FSL and compare them with the prior approaches in Table 3. Concretely, the *Linear* and *Mean-centroid* denote FT and NNC baselines, respectively, presented in Section 3.2. The *Ft-CC* denotes fine-tuning a cosine classifier based on the frozen feature extractor. It is obvious that the proposed approach can surpass all three transfer learning baselines by a large margin. For instance, *LAMR* performs better than the *Linear* by 5.4%, 4.6%, and 3.1% in {5/20/50}-shot settings, respectively. Similar to the observation in the multi-source FSL, *LAMR*++ also yields notable and consistent improvements over LAMR. Particularly on the ISIC dataset, *LAMR*++ can produce improvements of 5.5%, 5.8%, and 5.2% over our *LAMR* in {5/20/50}-shot settings, respectively.

Besides, we also make comparisons with other state-of-the-art methods [32,38,39,74]. LDP-net [32] imposes local-global feature consistency of prototypical networks by knowledge distillation, which improves the cross-domain generalization of the learned features. However, due to a lack of feature adaptation, LDP-net is typically inferior to other adaptation methods [38,39,74] and ours, especially for 20/50-shot settings. For example, on the ISIC dataset, *LAMR*++ performs significantly better than LDP-net by 6.0%, 9.8%, and 9.9% in {5/20/50}-shot settings, respectively.

Other methods [38,39,74] conduct domain-specific feature adaptation for tackling the large domain shift. Particularly, *FN* [38] adapts the feature extractor by fine-tuning its scaling and shifting parameters of batch normalization on few-shot data. We can observe that the *FN* is inferior to the transfer learning baselines in several cases of 5-shot settings. A possible interpretation for this phenomenon is that, with too limited data and extreme domain shift, optimizing the BN parameters accurately may be particularly hard. *ConFeSS* [39] proposes to learn a task-specific feature masking module that can produce refined features for further fine-tuning a target classifier and the feature extractor. *NSAE* [74] pretrains and fine-tunes the network with an additional auto-encoder to improve the model generalization, which implicitly augments the support data. Unlike *ConFeSS* [39] and *NSAE* [74], which leverage auxiliary modules for model adaptation, our approach is more efficient by directly optimizing the target model. Finally, we can also observe that our approach can achieve the highest accuracy among all the methods and experimental settings, except for one case in 5-way 5-shot EuroSAT where ConFeSS slightly outperforms our LAMR++ by 0.03%. But in 5-way 20-shot and 50-shot EuroSAT, our LAMR++ can achieve significant performance gains of 2.6% and 3.1% over it, respectively.

Table 3. The results of single-source few-shot learning on the BSCD-FSL benchmark. The best result in each setting is marked in bold.

Methods	ChestX			ISIC		
	5-Way 5-Shot	**5-Way 20-Shot**	**5-Way 50-Shot**	**5-Way 5-Shot**	**5-Way 20-Shot**	**5-Way 50-Shot**
ProtoNet [1] [9]	24.05 ± 1.01	28.21 ± 1.15	29.32 ± 1.12	39.57 ± 0.57	49.50 ± 0.55	51.99 ± 0.52
Linear [1] [13]	25.97 ± 0.41	31.32 ± 0.45	35.49 ± 0.45	48.11 ± 0.64	59.31 ± 0.48	66.48 ± 0.56
Mean-centroid [1] [75]	26.31 ± 0.42	30.41 ± 0.46	34.68 ± 0.46	47.16 ± 0.54	56.40 ± 0.53	61.57 ± 0.66
Ft-CC [1] [62]	26.95 ± 0.44	32.07 ± 0.55	34.76 ± 0.55	48.01 ± 0.49	58.13 ± 0.48	62.03 ± 0.52
FN [38]	25.78 ± 0.42	31.88 ± 0.46	34.81 ± 0.49	45.34 ± 0.60	58.92 ± 0.57	65.90 ± 0.58
ConFeSS [39]	27.09 ± 0.24	33.57 ± 0.31	39.02 ± 0.12	48.85 ± 0.29	60.10 ± 0.33	65.34 ± 0.45
NSAE [74]	27.10 ± 0.44	35.20 ± 0.48	38.95 ± 0.70	54.05 ± 0.63	66.17 ± 0.59	71.32 ± 0.61
LDP-net [2] [32]	27.30 ± 0.43	34.03 ± 0.49	37.58 ± 0.48	48.15 ± 0.60	58.47 ± 0.56	64.20 ± 0.55
LAMR	27.66 ± 0.44	33.82 ± 0.50	38.92 ± 0.50	48.66 ± 0.60	62.38 ± 0.60	68.92 ± 0.56
LAMR++	**28.86 ± 0.45**	**35.86 ± 0.50**	**41.36 ± 0.56**	**54.11 ± 0.62**	**68.22 ± 0.57**	**74.12 ± 0.54**
Methods	**EuroSAT**			**CropDiseases**		
	5-Way 5-Shot	**5-Way 20-Shot**	**5-Way 50-Shot**	**5-Way 5-Shot**	**5-Way 20-Shot**	**5-Way 50-Shot**
ProtoNet [1] [9]	73.29 ± 0.71	82.27 ± 0.57	80.48 ± 0.57	79.72 ± 0.67	88.15 ± 0.51	90.81 ± 0.43
Linear [1] [13]	79.08 ± 0.61	87.64 ± 0.47	91.34 ± 0.37	89.25 ± 0.51	95.51 ± 0.31	97.68 ± 0.21
Mean-centroid [1] [75]	82.21 ± 0.49	87.62 ± 0.34	88.24 ± 0.29	87.61 ± 0.47	93.87 ± 0.68	94.77 ± 0.34
Ft-CC [1] [62]	81.37 ± 1.54	86.83 ± 0.43	88.83 ± 0.38	89.15 ± 0.51	93.96 ± 0.46	94.27 ± 0.41
FN [38]	80.03 ± 0.70	88.94 ± 0.46	92.34 ± 0.36	91.11 ± 0.49	96.62 ± 0.26	98.27 ± 0.17
ConFeSS [39]	**84.65** ± 0.38	90.40 ± 0.24	92.66 ± 0.36	88.88 ± 0.51	95.34 ± 0.48	97.56 ± 0.43
NSAE [74]	83.96 ± 0.57	92.38 ± 0.33	95.42 ± 0.34	93.14 ± 0.47	98.30 ± 0.19	99.25 ± 0.14
LDP-net [2] [32]	81.50 ± 0.65	88.15 ± 0.48	90.75 ± 0.41	89.00 ± 0.51	95.49 ± 0.29	97.28 ± 0.20
LAMR	84.46 ± 0.55	92.21 ± 0.33	94.46 ± 0.27	94.15 ± 0.39	98.19 ± 0.17	99.16 ± 0.11
LAMR++	84.62 ± 0.55	**93.08 ± 0.31**	**95.75 ± 0.23**	**94.30 ± 0.38**	**98.39 ± 0.16**	**99.26 ± 0.11**

[1] the results from [21]. [2] reproduced results based the official released code [32] and their trained model.

6. Ablation Study and Analysis

We also conduct the ablation study based on the BSCD-FSL benchmark, as its target domains include a vast array of different specific recognition scenarios.

6.1. Effect of Multi-Domain Learning Framework

We first explore how our multi-source framework benefits feature transferability. We compare our framework with two baseline models: (1) *Single-source*: The feature representation is trained on one source dataset (that is *mini*-ImageNet). (2) *Merged-multi-sources*: The feature representation is trained by one task that classifies merged classes from all source datasets, as presented in Section 3. We validate the transferability of these representations based on the *NNC* baseline directly, and the results are reported in Table 4. We observe that the representation trained with *Merged-multi-sources* does better than the representation trained with *Sigle-source* in most cases. This is because the multi-source could provide a substantial amount of training data. Compared to *Merged-multi-sources*, our multi-source framework achieves much higher accuracy in most settings. Only for the ChestX, we see our framework is slightly underperforming within 1%. We conjecture that because the distribution of this target domain does not match any of the train distributions, thus robust knowledge transfer can not be ensured. However, the overall performance still demonstrates the benefit of using our multi-source representations rather than using one representation learned on the combined source dataset.

Table 4. Comparing transferability of feature representations trained by different models. The best result in each setting is marked in bold.

Methods	ChestX			ISIC		
	5-Way 5-Shot	5-Way 20-Shot	5-Way 50-Shot	5-Way 5-Shot	5-Way 20-Shot	5-Way 50-Shot
Single-source	25.90 ± 0.41	30.16 ± 0.45	32.76 ± 0.45	43.84 ± 0.55	51.98 ± 0.57	54.34 ± 0.53
Merged-multi-sources	**26.08 ± 0.41**	**31.14 ± 0.43**	**33.54 ± 0.45**	43.35 ± 0.55	51.71 ± 0.58	54.34 ± 0.53
Our Framework	25.96 ± 0.44	30.21 ± 0.43	32.58 ± 0.43	**48.61 ± 0.60**	**58.13 ± 0.59**	**60.54 ± 0.57**
Methods	**EuroSAT**			**CropDiseases**		
	5-Way 5-Shot	**5-Way 20-Shot**	**5-Way 50-Shot**	**5-Way 5-Shot**	**5-Way 20-Shot**	**5-Way 50-Shot**
Single-source	78.64 ± 0.61	84.05 ± 0.54	85.03 ± 0.48	88.27 ± 0.59	92.57 ± 0.44	94.19 ± 0.34
Merged-multi-sources	81.01 ± 0.56	86.05 ± 0.48	87.30 ± 0.41	90.22 ± 0.54	93.97 ± 0.36	95.09 ± 0.32
Our Framework	**84.76 ± 0.51**	**89.36 ± 0.41**	**89.88 ± 0.39**	**91.89 ± 0.50**	**95.22 ± 0.33**	**96.03 ± 0.27**

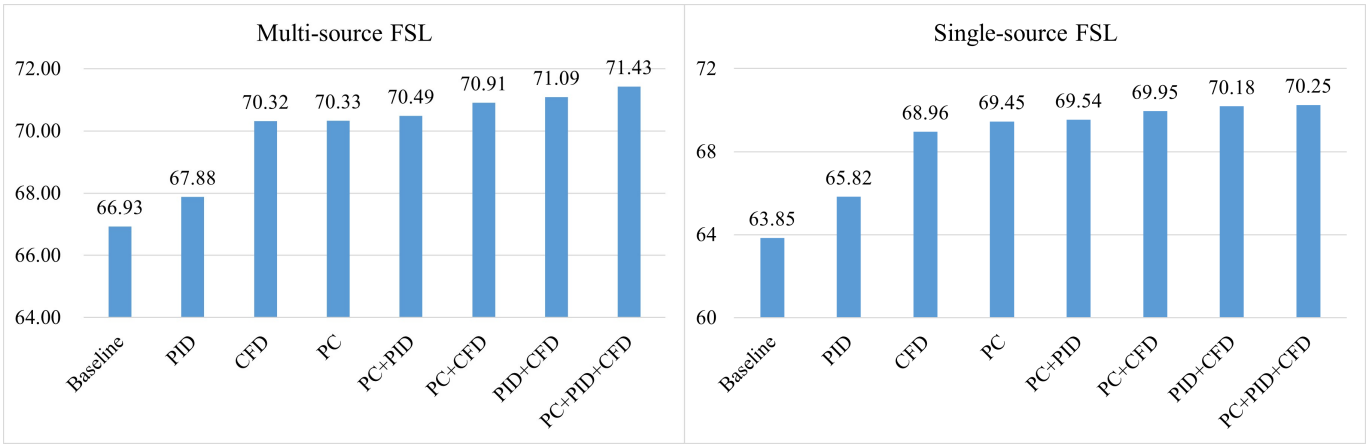
6.2. Significance of Few-Shot Adaptation

In order to understand what enables good representation adaptation over few-shot data, we systematically study the effect of all components in our adaptation loss, i.e., PID, CFD, PC with respect to Equations (11), (13) and (17). Table 5 shows the detailed results for all 24 settings that vary in two source types, four target domains, and three shot levels. We can make the following observations: (1) Applying any one of PID, CFD, or PC can lead to consistent performance gains in all 24 experimental settings. (2) With the combined supervision of PID and CFD, the results can be better than when only PID or CFD is used in 16 out of 24 settings. (3) Incorporating all three components can lead to the best result in 17 out of 24 settings. Particularly for the ISIC dataset with 50 examples per class available, the overall performance improves by up to 10.0% and 15.8% in multi- and single-source settings, respectively. It also verifies that our adaptation strategy is advantageous for dealing with the domain shift problem in few-shot learning. (4) The overall performance gains are more significant when more data are available. For example on the ChestX dataset, our adaptation method can yield improvements of {1.4%,4.0%,6.6%} over the baseline in {5/20/50}-shot settings, respectively. This also indicates that when suffering from extreme domain bias, such as in the medical domain, recognition requires more data to ensure good adaptation.

Table 5. Ablation study on 5-way K-shot performance by validating three components of the adaptation objective, including PID (parametric instance discrimination), CFD (class feature discrimination), and PC (prototypical classification). The best result in each setting is marked in bold.

K-Shot	PID	CFD	PC	Multi-Source FSL				Single-Source FSL			
				ChestX	ISIC	EuroSAT	CropDiseases	ChestX	ISIC	EuroSAT	CropDiseases
5				25.96 ± 0.44	48.61 ± 0.60	84.76 ± 0.51	91.89 ± 0.50	26.12 ± 0.42	42.96 ± 0.56	80.10 ± 0.62	89.38 ± 0.55
	✓			26.89 ± 0.45	50.61 ± 0.61	85.06 ± 0.51	92.64 ± 0.44	26.96 ± 0.43	45.12 ± 0.56	81.33 ± 0.61	91.72 ± 0.48
		✓		26.84 ± 0.41	51.92 ± 0.63	**87.12** ± 0.47	94.51 ± 0.40	26.96 ± 0.44	48.22 ± 0.61	84.36 ± 0.55	93.94 ± 0.41
			✓	27.06 ± 0.42	52.14 ± 0.61	86.42 ± 0.48	94.20 ± 0.41	27.49 ± 0.43	47.78 ± 0.59	83.50 ± 0.57	93.40 ± 0.43
	✓		✓	27.17 ± 0.42	52.12 ± 0.61	86.50 ± 0.48	94.19 ± 0.41	27.43 ± 0.42	47.90 ± 0.59	83.54 ± 0.56	93.48 ± 0.42
		✓	✓	27.39 ± 0.42	52.43 ± 0.61	86.91 ± 0.47	**94.62** ± 0.39	27.56 ± 0.44	48.54 ± 0.59	84.42 ± 0.55	94.15 ± 0.39
	✓	✓		27.35 ± 0.46	52.53 ± 0.64	86.77 ± 0.47	94.48 ± 0.38	27.55 ± 0.44	48.59 ± 0.59	84.32 ± 0.55	94.05 ± 0.40
	✓	✓	✓	**27.37** ± 0.41	**52.58** ± 0.61	86.92 ± 0.47	94.61 ± 0.39	**27.66** ± 0.44	**48.66** ± 0.60	**84.46** ± 0.55	**94.15** ± 0.39
20				30.21 ± 0.43	58.13 ± 0.59	89.36 ± 0.41	95.22 ± 0.33	30.92 ± 0.43	50.41 ± 0.57	84.78 ± 0.53	93.73 ± 0.40
	✓			31.60 ± 0.46	58.87 ± 0.58	89.87 ± 0.40	95.83 ± 0.27	32.22 ± 0.45	53.49 ± 0.56	86.45 ± 0.51	95.42 ± 0.33
		✓		30.95 ± 0.43	63.90 ± 0.60	93.65 ± 0.29	98.24 ± 0.18	31.32 ± 0.44	61.09 ± 0.64	91.99 ± 0.34	98.01 ± 0.18
			✓	33.45 ± 0.46	63.96 ± 0.57	92.97 ± 0.31	97.83 ± 0.20	33.20 ± 0.48	61.09 ± 0.59	91.49 ± 0.36	97.80 ± 0.19
	✓		✓	33.33 ± 0.48	64.01 ± 0.56	92.99 ± 0.31	97.84 ± 0.20	33.33 ± 0.48	61.16 ± 0.59	91.46 ± 0.36	97.78 ± 0.19
		✓	✓	33.64 ± 0.47	65.19 ± 0.57	93.60 ± 0.29	98.26 ± 0.18	33.64 ± 0.47	62.13 ± 0.61	92.18 ± 0.33	98.20 ± 0.17
	✓	✓		**34.26** ± 0.49	64.69 ± 0.56	93.52 ± 0.28	98.15 ± 0.16	**34.06** ± 0.49	62.29 ± 0.60	92.21 ± 0.34	98.07 ± 0.18
	✓	✓	✓	34.16 ± 0.48	**65.33** ± 0.55	**93.65** ± 0.29	**98.26** ± 0.18	33.82 ± 0.50	**62.38** ± 0.60	**92.21** ± 0.33	**98.19** ± 0.17
50				32.58 ± 0.43	60.54 ± 0.57	89.88 ± 0.39	96.03 ± 0.27	33.87 ± 0.46	53.15 ± 0.54	85.71 ± 0.50	95.06 ± 0.30
	✓			34.69 ± 0.47	61.18 ± 0.54	90.63 ± 0.38	96.69 ± 0.23	36.01 ± 0.46	56.70 ± 0.55	87.80 ± 0.45	96.67 ± 0.24
		✓		33.47 ± 0.43	68.51 ± 0.68	**95.48** ± 0.23	**99.19** ± 0.11	34.53 ± 0.44	63.74 ± 0.77	94.26 ± 0.28	99.11 ± 0.11
			✓	36.59 ± 0.48	66.46 ± 0.56	94.37 ± 0.26	98.50 ± 0.16	38.50 ± 0.50	66.56 ± 0.56	93.86 ± 0.28	98.77 ± 0.13
	✓		✓	37.61 ± 0.49	67.05 ± 0.55	94.47 ± 0.26	98.56 ± 0.15	38.44 ± 0.51	67.20 ± 0.55	93.88 ± 0.28	98.84 ± 0.13
		✓	✓	34.91 ± 0.48	69.40 ± 0.58	95.45 ± 0.23	99.10 ± 0.12	37.24 ± 0.47	67.71 ± 0.60	94.44 ± 0.27	99.16 ± 0.11
	✓	✓		37.97 ± 0.52	69.08 ± 0.55	95.23 ± 0.23	99.03 ± 0.10	**39.07** ± 0.48	68.43 ± 0.56	94.44 ± 0.26	99.09 ± 0.11
	✓	✓	✓	**39.21** ± 0.51	**70.52** ± 0.53	95.42 ± 0.23	99.12 ± 0.11	38.92 ± 0.50	**68.92** ± 0.56	**94.46** ± 0.27	**99.16** ± 0.11

To better evaluate the effectiveness of each isolated component and different combinations of the three components towards performance in cross-domain few-shot transfer, we further compute the average accuracy across different datasets and shot levels and rank them in Figure 9. We can observe that the rank of isolated performance gains of using PID, CFD, or PC is {*PC* > *CFD* > *PID*} for both multi-source and single-source FSL. The full adaptation can achieve the best mean accuracy in both multi-source and single-source settings. Overall, the adaptation provides an average improvement of 4.5% and 6.4% for multi-source FSL and single-source FSL, respectively.



(**a**) Mean accuracy on Multi-source FSL (**b**) Mean accuracy on Single-source FSL

Figure 9. Evaluating the effectiveness of each isolated component and different combinations of the three components (PID, CFD, and PC) towards performance in cross-domain few-shot transfer. (**a**) Mean accuracy across different datasets and shot levels for multi-source FSL. (**b**) Mean accuracy across different datasets and shot levels for single-source FSL. We can observe that the full adaptation can achieve the best mean accuracy in both multi-source and single-source settings.

6.3. Effect of Different Classifier Learning

We also compare some variants built on our pre-trained multi-source representations using different classification modules or fine-tuning regimes:

- *Fixed-MSR*: Directly leveraging the frozen multi-source representations with *NNC* baseline.
- *Ft-LC*: Fine-tuning a linear classification layer on each frozen representation head.
- *Ft-CC*: Fine-tuning a cosine classification layer on each frozen representation head.
- *Ft-MSR-LC*: Fine-tuning both multi-source representations (projection layers) and following linear classification layers.
- *Ft-MSR-CC*: Fine-tuning both multi-source representations (projection layers) and following cosine classification layers.

The results are reported in Table 6. We can observe that: (1) only fine-tuning a classifier on the frozen representations can obtain performance gains in all settings. Besides, fine-tuning cosine classifiers (*Ft-CC*) always outperform linear classifier-based ones (*Ft-LC*), which is also consistent with the previous literature findings [13,21]. (2) further adapting the representations associated with the classifiers (Ft-MSR-LC & Ft-MSR-CC) can also lead to accuracy boosts.

Table 6. Quantitative analysis of different classifiers that are incorporated into our pre-trained multi-source representations during the meta-test stage.

Methods	ChestX			ISIC		
	5-Way 5-Shot	**5-Way 20-Shot**	**5-Way 50-Shot**	**5-Way 5-Shot**	**5-Way 20-Shot**	**5-Way 50-Shot**
Fixed-MSR	25.96 ± 0.44	30.20 ± 0.46	32.58 ± 0.46	48.61 ± 0.62	58.13 ± 0.61	60.54 ± 0.57
Ft-LC	25.68 ± 0.44	30.53 ± 0.46	33.64 ± 0.48	51.32 ± 0.63	61.52 ± 0.57	64.18 ± 0.56
Ft-CC	26.75 ± 0.44	32.69 ± 0.48	37.19 ± 0.53	51.51 ± 0.64	63.59 ± 0.57	67.75 ± 0.55
Ft-MSR-LC	26.25 ± 0.45	31.10 ± 0.46	34.26 ± 0.48	51.81 ± 0.62	62.56 ± 0.57	65.07 ± 0.56
Ft-MSR-CC	27.04 ± 0.45	33.31 ± 0.49	38.26 ± 0.52	52.12 ± 0.63	64.52 ± 0.56	70.00 ± 0.53
Methods	**EuroSAT**			**CropDiseases**		
	5-Way 5-Shot	**5-Way 20-Shot**	**5-Way 50-Shot**	**5-Way 5-Shot**	**5-Way 20-Shot**	**5-Way 50-Shot**
Fixed-MSR	84.76 ± 0.51	89.36 ± 0.40	89.88 ± 0.39	91.89 ± 0.47	95.22 ± 0.29	96.03 ± 0.25
Ft-LC	85.28 ± 0.48	91.01 ± 0.35	91.85 ± 0.32	92.85 ± 0.42	96.78 ± 0.22	97.64 ± 0.16
Ft-CC	86.71 ± 0.48	93.08 ± 0.29	94.52 ± 0.25	94.11 ± 0.39	97.93 ± 0.17	98.83 ± 0.11
Ft-MSR-LC	85.82 ± 0.48	91.62 ± 0.33	92.48 ± 0.30	93.51 ± 0.40	97.21 ± 0.20	98.00 ± 0.15
Ft-MSR-CC	86.77 ± 0.48	93.48 ± 0.28	95.09 ± 0.23	94.45 ± 0.38	98.18 ± 0.16	99.09 ± 0.10

6.4. Impressive Visualization

To further qualitatively understand how the adaptation leads to few-shot performance gains, we visualize feature embeddings of the query images and the class prototypes by t-SNE [76] in Figure 10, which is computed in a 5-way 5-shot task sampled from the CropDisease dataset. Figure 10a–f denote the multi-head representations, each of which shows its feature embeddings before or after the adaptation, respectively. The benefits of our adaptation method can be attributed to two aspects: (1) The query features of the same class become more compact, and the class clusters are more separable from each other after making the adaptation. It also indicates that our LAMR can encourage intra-class compactness and inter-class divergence, which results in more discriminative features for classification. (2) The induced class prototypes computed from the adapted instance proxy are also more representative so the prototypes can be used to classify the query features well. These observations can also explain the significant performance boosts presented in the ablation study.

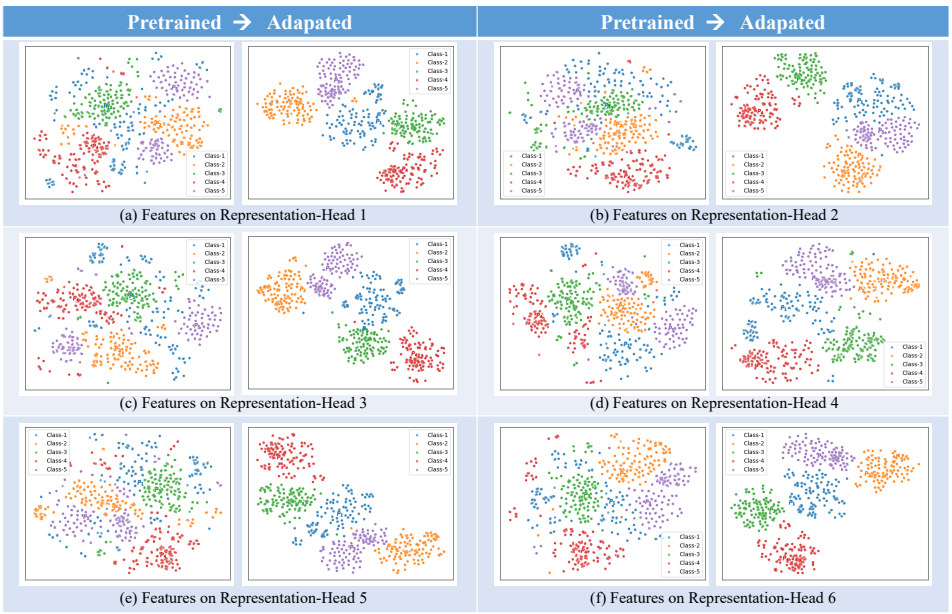


Figure 10. The t-SNE visualization of the feature distribution on the multi-head representations before and after making adaptation by our method. (**a**–**f**) show the six representation spaces learned on the BSCD-FSL benchmark. Better viewed in color with zoom-in.

We further analyze how features are changed with the adaptation by visualizing class activation maps (CAMs) [77] on a 5-shot task sampled from the CropDisease, which can also be regarded as a fine-grained recognition task on grape diseases. Comparisons of regions that the deep CNNs focus on for discrimination before and after the adaptation are shown in Figure 11. We can make the following observations: (1) For the healthy grape leaf, visual cues to make predictions do not significantly change before and after the adaptation, which indicates that the pre-trained features can be generalized enough to recognize such common natural objects. However, we can still see that the adaptation can make CNN features more focused on the skeleton and edge of the leaf but less on the background. (2) For the two grape leaf diseases, our adaptation method can help CNN concentrate on the most relevant regions with respect to the two specific diseases. In contrast, the CNN features without the adaptation still focus on the visual cues of the common object, such as the leaf edge. The change indicates our method can improve discrimination towards the class-specific visual cues. The ability to steer the task-relevant features verifies the superior performance of our LAMR.

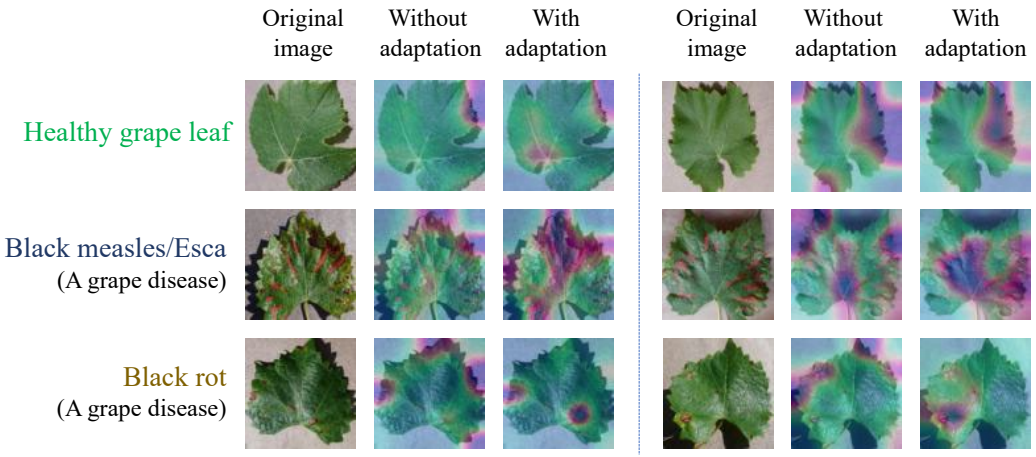


Figure 11. Visualization using the class activation maps (CAMs) to show the regions that deep networks focus on before and after the adaptation. Image examples from a 5-shot task in the CropDisease. Better viewed in color.

7. Conclusions and Future Work

In this paper, we investigate a more practical FSL setting, namely multi-source cross-domain few-shot learning. To tackle the problem, we propose a simple yet effective multi-source representation framework for learning prior knowledge from multiple datasets, which enables generalization to a wide range of unseen domains. Further task-specific adaptation on few-shot data is performed to enhance instance discrimination and class discrimination by minimizing two contrastive losses on the multi-domain representations. We empirically demonstrate the superiority of our LAMR over many previous methods and strong baselines, which achieves state-of-the-art results for cross-domain FSL. We extend LAMR to single-source FSL by introducing dataset-splitting strategies that equally split one source dataset into sub-domains. The empirical results show that applying simple "random splitting" can improve conventional cosine-similarity-based classifiers in FSL with a fixed single-source data budget. Extensive ablation studies and analyses illustrate that each component of our method can effectively facilitate few-shot transfer. Our method also has some limitations, and we could see some promising future directions. First, we conduct adaptation by either fine-tuning or freezing the full backbone. It would be promising for future work to seek more flexible adaptation methods that can select a part of layers or parameters to adjust conditioning on the given task. Second, a study [65] has also shown that the choice of the source training dataset has a huge impact on the performance of downstream tasks. We also acknowledge that not every dataset in the multi-source domains contributes equally to the target task. Further improvements can also be explored for a

more scalable transfer by considering the similarity between the source and target domains. It is worth noting that the two limitations may also exist with most other methods that focus on representation learning on source data or adaptation on few-shot data. Besides, to our knowledge, multi-source few-shot learning on other fundamental computer vision applications, such as segmentation and detection has not been explored yet. Developing new benchmarks for those computer vision problems would also foster future progress in this field.

Author Contributions: Conceptualization, G.L.; methodology, G.L.; software, G.L. and Z.Z.; validation, G.L. and Z.Z.; investigation, G.L. and Z.Z.; writing—original draft preparation, G.L.; writing—review and editing, G.L., Z.Z. and X.F.; supervision, X.F.; project administration, X.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data used in this study are publicly available.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. Acm* **2017**, *60*, 84–90. [CrossRef]
2. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016 ; pp. 770–778.
3. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
4. Fei-Fei, L.; Fergus, R.; Perona, P. One-shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 594–611. [CrossRef] [PubMed]
5. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Montreal, QC, Canada, 8–13 December 2014; pp. 3320–3328.
6. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
7. Finn, C.; Abbeel, P.; Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 1126–1135.
8. Ravi, S.; Larochelle, H. Optimization as a model for few-shot learning. In Proceedings of the International Conference on Learning Representations, San Juan, Puerto Rico, 2–4 May 2016.
9. Snell, J.; Swersky, K.; Zemel, R. Prototypical Networks for Few-shot Learning. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4077–4087.
10. Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; Wierstra, D. Matching Networks for One Shot Learning. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 3630–3638.
11. Thrun, S. Lifelong learning algorithms. In *Learning to Learn*; Springer: Berlin/Heidelberg, Germany, 1998; pp. 181–209.
12. Chen, Y.; Liu, Z.; Xu, H.; Darrell, T.; Wang, X. Meta-Baseline: Exploring Simple Meta-Learning for Few-Shot Learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 11–17 October 2021; pp. 9062–9071.
13. Chen, W.Y.; Liu, Y.C.; Kira, Z.; Wang, Y.C.F.; Huang, J.B. A Closer Look at Few-shot Classification. In Proceedings of the International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6–9 May 2019.
14. Tian, Y.; Wang, Y.; Krishnan, D.; Tenenbaum, J.B.; Isola, P. Rethinking few-shot image classification: A good embedding is all you need? In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020.
15. Wang, Y.; Chao, W.L.; Weinberger, K.Q.; van der Maaten, L. SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning. *arXiv* **2019**, arXiv:1911.04623.
16. Dhillon, G.S.; Chaudhari, P.; Ravichandran, A.; Soatto, S. A Baseline for Few-Shot Image Classification. In Proceedings of the International Conference on Learning Representations (ICLR), Addis Ababa, Ethiopia, 30 April 2020.
17. Raghu, A.; Raghu, M.; Bengio, S.; Vinyals, O. Rapid learning or feature reuse? towards understanding the effectiveness of maml. In Proceedings of the International Conference on Learning Representations (ICLR), Addis Ababa, Ethiopia, 30 April 2020.
18. Gidaris, S.; Bursuc, A.; Komodakis, N.; Perez, P.; Cord, M. Boosting Few-Shot Visual Learning with Self-Supervision. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019.
19. Afrasiyabi, A.; Lalonde, J.F.; Gagné, C. Associative Alignment for Few-shot Image Classification. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020.
20. Oreshkin, B.; Rodríguez López, P.; Lacoste, A. TADAM: Task dependent adaptive metric for improved few-shot learning. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 721–731.

21. Guo, Y.; Codella, N.C.; Karlinsky, L.; Codella, J.V.; Smith, J.R.; Saenko, K.; Rosing, T.; Feris, R. A broader study of cross-domain few-shot learning. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 124–141.
22. Triantafillou, E.; Zhu, T.; Dumoulin, V.; Lamblin, P.; Evci, U.; Xu, K.; Goroshin, R.; Gelada, C.; Swersky, K.; Manzagol, P.A.; et al. Meta-Dataset: A Dataset of Datasets for Learning to Learn from Few Examples. In Proceedings of the International Conference on Learning Representations (ICLR), Addis Ababa, Ethiopia, 30 April 2020.
23. Helber, P.; Bischke, B.; Dengel, A.; Borth, D. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, *12*, 2217–2226. [CrossRef]
24. Tschandl, P.; Rosendahl, C.; Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Sci. Data* **2018**, *5*, 180161. [CrossRef] [PubMed]
25. Codella, N.; Rotemberg, V.; Tschandl, P.; Celebi, M.E.; Dusza, S.; Gutman, D.; Helba, B.; Kalloo, A.; Liopyris, K.; Marchetti, M.; et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv* **2019**, arXiv:1902.03368.
26. Wang, X.; Peng, Y.; Lu, L.; Lu, Z.; Bagheri, M.; Summers, R.M. 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 (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2097–2106.
27. Dvornik, N.; Schmid, C.; Mairal, J. Selecting relevant features from a multi-domain representation for few-shot classification. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 769–786.
28. Liu, L.; Hamilton, W.L.; Long, G.; Jiang, J.; Larochelle, H. A Universal Representation Transformer Layer for Few-Shot Image Classification. In Proceedings of the International Conference on Learning Representations (ICLR), Virtual Event, 3–7 May 2021.
29. Li, W.H.; Liu, X.; Bilen, H. Universal representation learning from multiple domains for few-shot classification. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 9526–9535.
30. Liu, G.; Zhang, Z.; Cai, F.; Liu, D.; Fang, X. Learning and Adapting Diverse Representations for Cross-domain Few-shot Learning. In Proceedings of the 2023 IEEE International Conference on Data Mining Workshops (ICDMW), Shanghai, China, 1–4 December 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 294–303.
31. Bontonou, M.; Béthune, L.; Gripon, V. Predicting the generalization ability of a few-shot classifier. *Information* **2021**, *12*, 29. [CrossRef]
32. Zhou, F.; Wang, P.; Zhang, L.; Wei, W.; Zhang, Y. Revisiting prototypical network for cross domain few-shot learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 20061–20070.
33. Zhao, L.; Liu, G.; Guo, D.; Li, W.; Fang, X. Boosting Few-shot visual recognition via saliency-guided complementary attention. *Neurocomputing* **2022**, *507*, 412–427. [CrossRef]
34. Liu, C.; Fu, Y.; Xu, C.; Yang, S.; Li, J.; Wang, C.; Zhang, L. Learning a few-shot embedding model with contrastive learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 8635–8643.
35. Rebuffi, S.A.; Bilen, H.; Vedaldi, A. Efficient parametrization of multi-domain deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 8119–8127.
36. Perez, E.; Strub, F.; De Vries, H.; Dumoulin, V.; Courville, A. Film: Visual reasoning with a general conditioning layer. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
37. Lifchitz, Y.; Avrithis, Y.; Picard, S.; Bursuc, A. Dense Classification and Implanting for Few-Shot Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
38. Yazdanpanah, M.; Rahman, A.A.; Chaudhary, M.; Desrosiers, C.; Havaei, M.; Belilovsky, E.; Kahou, S.E. Revisiting Learnable Affines for Batch Norm in Few-Shot Transfer Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 9109–9118.
39. Das, D.; Yun, S.; Porikli, F. ConfeSS: A framework for single source cross-domain few-shot learning. In Proceedings of the International Conference on Learning Representations (ICLR), Virtual Event, 25–29 April 2022.
40. Li, W.H.; Liu, X.; Bilen, H. Cross-domain Few-shot Learning with Task-specific Adapters. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 7161–7170.
41. Liu, G.; Zhao, L.; Fang, X. PDA: Proxy-based domain adaptation for few-shot image recognition. *Image Vis. Comput.* **2021**, *110*, 104164. [CrossRef]
42. Soudy, M.; Afify, Y.M.; Badr, N. GenericConv: A Generic Model for Image Scene Classification Using Few-Shot Learning. *Information* **2022**, *13*, 315. [CrossRef]
43. Csányi, G.M.; Vági, R.; Megyeri, A.; Fülöp, A.; Nagy, D.; Vadász, J.P.; Üveges, I. Can Triplet Loss Be Used for Multi-Label Few-Shot Classification? A Case Study. *Information* **2023**, *14*, 520. [CrossRef]
44. Cai, J.; Wu, L.; Wu, D.; Li, J.; Wu, X. Multi-Dimensional Information Alignment in Different Modalities for Generalized Zero-Shot and Few-Shot Learning. *Information* **2023**, *14*, 148. [CrossRef]
45. Tzeng, E.; Hoffman, J.; Zhang, N.; Saenko, K.; Darrell, T. Deep Domain Confusion: Maximizing for Domain Invariance. *arXiv* **2014**, arXiv:1412.3474.

46. Long, M.; Cao, Y.; Wang, J.; Jordan, M. Learning transferable features with deep adaptation networks. In Proceedings of the International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015; pp. 97–105.
47. Ganin, Y.; Lempitsky, V. Unsupervised domain adaptation by backpropagation. In Proceedings of the International Conference on Machine Learning (ICML), Lille, France, 7–9 July 2015; pp. 1180–1189.
48. Peng, X.; Bai, Q.; Xia, X.; Huang, Z.; Saenko, K.; Wang, B. Moment matching for multi-source domain adaptation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1406–1415.
49. Xu, R.; Chen, Z.; Zuo, W.; Yan, J.; Lin, L. Deep cocktail network: Multi-source unsupervised domain adaptation with category shift. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 3964–3973.
50. Wu, Z.; Xiong, Y.; Yu, S.X.; Lin, D. Unsupervised Feature Learning via Non-Parametric Instance Discrimination. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
51. Oord, A.v.d.; Li, Y.; Vinyals, O. Representation learning with contrastive predictive coding. *arXiv* **2018**, arXiv:1807.03748.
52. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 9729–9738.
53. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In Proceedings of the International Conference on Machine Learning (ICML), Virtual Event, 13–18 July 2020; pp. 1597–1607.
54. Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. Supervised contrastive learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 18661–18673.
55. Bilen, H.; Vedaldi, A. Universal representations: The missing link between faces, text, planktons, and cat breeds. *arXiv* **2017**, arXiv:1701.07275.
56. Guo, Y.; Li, Y.; Wang, L.; Rosing, T. Depthwise convolution is all you need for learning multiple visual domains. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 29–31 January 2019; Volume 33, pp. 8368–8375.
57. Dvornik, N.; Schmid, C.; Mairal, J. Diversity with Cooperation: Ensemble Methods for Few-Shot Classification. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019.
58. Chen, Z.; Badrinarayanan, V.; Lee, C.Y.; Rabinovich, A. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In Proceedings of the International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018; pp. 794–803.
59. Liu, G.; Zhao, L.; Li, W.; Guo, D.; Fang, X. Class-wise Metric Scaling for Improved Few-Shot Classification. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Virtual, 5–9 January 2021; pp. 586–595.
60. Yu, C.; Zhao, X.; Zheng, Q.; Zhang, P.; You, X. Hierarchical Bilinear Pooling for Fine-Grained Visual Recognition. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
61. Kim, J.; On, K.W.; Lim, W.; Kim, J.; Ha, J.; Zhang, B. Hadamard Product for Low-rank Bilinear Pooling. In Proceedings of the International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017.
62. Gidaris, S.; Komodakis, N. Dynamic Few-Shot Visual Learning without Forgetting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
63. Qi, H.; Brown, M.; Lowe, D.G. Low-Shot Learning with Imprinted Weights. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
64. Lee, K.; Maji, S.; Ravichandran, A.; Soatto, S. Meta-learning with differentiable convex optimization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 10657–10665.
65. Sbai, O.; Couprie, C.; Aubry, M. Impact of base dataset design on few-shot image classification. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 597–613.
66. Wah, C.; Branson, S.; Welinder, P.; Perona, P.; Belongie, S. The Caltech-UCSD Birds-200-2011 Dataset. In *Technical Report CNS-TR-2011-001*; California Institute of Technology: Pasadena, CA, USA, 2011.
67. Krizhevsky, A. *Learning Multiple Layers of Features from Tiny Images*; Technical report; University of Toronto: Toronto, ON, Canada, 2009.
68. Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; Vedaldi, A. Describing textures in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 3606–3613.
69. Griffin, G.; Holub, A.; Perona, P. *Caltech-256 Object Category Dataset*; Technical Report; California Institute of Technology: Pasadena, CA, USA, 2007.
70. Mohanty, S.P.; Hughes, D.P.; Salathé, M. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **2016**, *7*, 1419. [CrossRef] [PubMed]
71. Liu, B.; Cao, Y.; Lin, Y.; Li, Q.; Zhang, Z.; Long, M.; Hu, H. Negative Margin Matters: Understanding Margin in Few-shot Classification. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020.
72. Yang, S.; Liu, L.; Xu, M. Free Lunch for Few-shot Learning: Distribution Calibration. In Proceedings of the International Conference on Learning Representations, Virtual Event, 3–7 May 2021.

73. dan Guo, D.; Tian, L.; Zhao, H.; Zhou, M.; Zha, H. Adaptive Distribution Calibration for Few-Shot Learning with Hierarchical Optimal Transport. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 6996–7010.
74. Liang, H.; Zhang, Q.; Dai, P.; Lu, J. 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, Montreal, BC, Canada, 11–17 October 2021; pp. 9424–9434.
75. Mensink, T.; Verbeek, J.; Perronnin, F.; Csurka, G. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2624–2637. [CrossRef] [PubMed]
76. Maaten, L.V.D.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
77. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.