Pattern Recognition 155 (2024) 110640

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Few-shot image classification via hybrid representation

Bao-Di Liu [a], Shuai Shao [b,*], Chunyan Zhao [c], Lei Xing [d], Weifeng Liu [a], Weijia Cao [e], Yicong Zhou [f]

[a] College of Control Science and Engineering, China University of Petroleum, Qingdao 266580, China
[b] Zhejiang Lab, Hangzhou, Zhejiang 311121, China
[c] Suzhou Centennial College, China
[d] Qingdao Chrystar Electronic Technology Co., Ltd, Qingdao 266580, China
[e] Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China
[f] Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, China

ARTICLE INFO

Keywords:
Few-shot image classification
Specific representation
Shared representation

ABSTRACT

Few-shot image classification aims to learn an embedding model on the base datasets and design a base learner to recognize novel categories. The few-shot image classification framework is a two-phase process. First, the pre-train phase utilizes the base data to train a CNN-based feature extractor. Next, in the meta-test phase, the frozen feature extractor is applied to novel data with categories different from the base data. A base learner is then designed for recognition. Several simple base learners, including nearest neighbor, support vector machine, and logistic regression classifiers, have been recently introduced for few-shot learning tasks. However, these base learners are separately designed to consider specific representations (e.g., the class center) or shared representations (e.g., the boundaries). This paper mainly focuses on exploring the representation-residual base learners, which aim to represent a query sample with the support set and predict the query sample's label based on the minimal residual error. We first introduce two representation-residual base learners: a specific representation base learner and a shared representation base learner. Then, we propose a novel hybrid representation base learner that combines both base learners to generate competitive representation. Additionally, we extend our approach by incorporating a self-training framework to utilize the query data fully. We evaluate our proposed method on several benchmark few-shot image classification datasets, such as miniImageNet, tieredImageNet, CIFAR-FS, FC100, and CUB datasets. The experimental results indicate that our proposed approach shows a significant performance improvement.

1. Introduction

Recently, deep learning has achieved impressive performance in various visual recognition tasks, such as object detection [1,2], image classification [3,4], or semantic segmentation [5]. This success typically relies on a large number of labeled datasets. However, deep learning typically relies on large labeled datasets, which can be costly to collect. This is in contrast to the human visual recognition system, which can learn a novel concept with only a few examples or a limited amount of experience. Therefore, a recently emerging approach called few-shot learning [6–11] has attracted increasing attention. Few-shot learning aims to build a base learner for a novel concept from very few labeled examples, making it a more cost-effective and efficient learning method.

Recent efforts to solve few-shot learning problems usually utilize learning-to-learn (i.e., meta-learning) approaches and the model-classifier decoupling method. Typically, the meta-learning strategy comprises an embedding model that maps the input images into a feature space and a base learner that associates the feature space with various tasks. Meta-learning models are trained by a large number of few-shot classification tasks that aim to make the base learner generalize well to the novel cases. The model-classifier decoupling strategy learns the embedding model without the base learner. Wang et al. [12] found that learning a robust feature model with a softmax layer was more effective than the complex meta-learning algorithm. Instead of extracting the meta-task training model during training, they used the classical neural network training method. During the meta-testing stage, remove the softmax layer of the neural network as the feature model and then use a robust classification to achieve excellent classification performance. Therefore, the training embedding models and designing base learners are equally important in the few-shot image classification.

[*] Corresponding author.
E-mail addresses: thu.liubaodi@gmail.com (B.-D. Liu), shaoshuai0914@gmail.com (S. Shao), zhaocy@scc.edu.cn (C. Zhao), upc_xl@163.com (L. Xing), liuwf@upc.edu.cn (W. Liu), caowj@aircas.ac.cn (W. Cao), yicongzhou@um.edu.mo (Y. Zhou).

https://doi.org/10.1016/j.patcog.2024.110640
Received 30 April 2022; Received in revised form 29 May 2023; Accepted 28 May 2024
Available online 30 May 2024

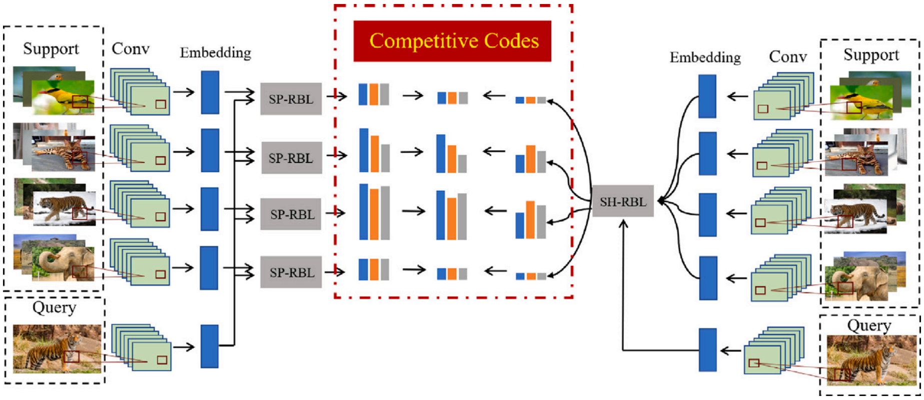


Fig. 1. The framework of HY-RBL. The left is the specific representation base learner (SP-RBL), and the right is the shared representation base learner (SH-RBL). For the SP-RBL, a query sample is represented by each specific class and would generate large weights for each support sample. It is reasonable if the label of the query sample is the same as the label of support samples (e.g., if the label of the query sample is tiger, it would be represented well by the support samples from the tiger category.) For the SH-RBL, a query sample is represented by all the support samples, and the weights for each support sample are different. The large weights would assign to the support samples similar to the query. SP-RBL would generate the optimal representation (minimal residual) for each specific class, while SH-RBL would generate the optimal representation (minimal residual) for all support samples. The HY-RBL combines SP-RBL and SH-RBL to generate competitive codes or representations. Concretely, the SP-RBL would prevent SH-RBL from generating imprecise representation for each specific class, while SH-RBL would suppress large weights generated from SP-RBL.

Base learner design is one of the fundamental issues in computer vision areas. Snell et al. [13] utilized the nearest feature center classifier to update the embedding model. Bertinetto et al. [14] learned a linear regression classifier to obtain a classifier plane. These base learners only consider the within-class information for each specific class. Lee et al. [15] proposed to use a linear support vector machine (SVM), which exploits the negative examples to learn class boundaries, as the base learner. Simon et al. [16] proposed an adaptive subspace classifier, which found a suitable subspace for each class, then measured the distance in the subspace and predicted the label. Shao et al. [17] proposed a multi-head feature collaboration method, which attempts to represent samples by fusing multi-head features collaboratively. This method helps strengthen the model's efficacy and robustness.

In this paper, we mainly focus on exploring representation-residual base learners for the model-classifier decoupling method. The designed base learners are shown in Fig. 1. First, we introduce the specific representation base learner (SP-RBL), which can be considered as representing the query sample in each individual subspace (the support samples in each specific class form a subspace). The query sample is well represented in each subspace. Second, we present the shared representation base learner (SH-RBL), which maps all the query samples in the same subspace. The query sample's representation usually assigns a large weight to its neighbor support samples. Third, we propose a novel hybrid representation base learner, combining the specific representation base learner and the shared representation base learner. It can combine the advantages of the specific representation base learner and the shared representation base learner, representing the query sample in the same subspace and holding each specific class's description ability to generate more discriminative representation. Moreover, we extend the self-training framework to our approach to fully utilize the query data. In summary, the main contributions are three-fold:

- We explore the representation-residual classifier and introduce two types of representation-residual base learners: specific representation-base learners and shared representation-base learners.
- We propose a novel hybrid representation base learner, which considers specific description and shared correction.
- We propose a self-training few-shot learning method and expand the self-training framework to our approach using query data. The introduction of self-training improves the model's generalizability.
- We show that our proposed approach has achieved state-of-the-art performance on several benchmark datasets compared with few-shot classification approaches.

2. Related work

The idea of *meta-learning* has been widely explored in many ways; these approaches for few-shot learning can be simply divided into three categories:

(*i*) Optimization-based methods. They aim to learn an automatic initialization parameter instead of a handcrafted one relying on training tasks. For instance, MAML [18] and Reptile [19] are the most typical ones in these methods. MAML proposed a model-agnostic algorithm that focused on the initialization for a variety of different learning problems by using gradient descent, while Reptile paid attention to the trained weights according to repeatedly sampling a task and training on it. Es-MAML [20] presents a new framework based on evolutionary strategy, which avoids the second derivative estimation problem and can handle novel non-smooth adaptive operators.

(*ii*) Black-box adaption-based methods. They pay attention to learning a neural network by training tasks. The parameters of the neural network always depend on the RNN-based model. Then the neural network would be properly harnessed in testing tasks. In Snail [21], they proposed a novel architecture to aggregate information for any period. And Ravi et al. [22] proposed a model based on an LSTM meta-learner to capture short-term and long-term knowledge.

(*iii*) Metric-based methods. The distance-based rules are exploited to compare the difference between different samples. For example, the samples in prototypical network [13] are mapped into the nearest-neighbor-based metric space. Different from the prototypical network, MetaOpt [15] uses the SVM base learner to separate samples of different classes into different subspaces. Then the reconstruction error is used to update the network. DeepEMD [23] split the image into multiple blocks, and then calculated the optimal matching cost between the query set and the image block of the support set using the distance of earth movement as the distance measure to represent the similarity. TDE [24] embeds dictionary learning methods into few-shot learning frameworks and maps feature embeds to more discriminative subspaces to suit specific tasks.

Our work is related to the metric-based method. We mainly focus on exploring the representation-residual base learner. We show that representation-residual base learner methods can effectively improve the performance of few-shot classification.

3. Problem setup

Let $\mathcal{X}$ be the inputs (e.g., images) and $\mathcal{Y}$ be the corresponding labels. Let $\mathcal{P}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$. Supervised machine learning algorithm typically aims to obtain a parameterized model $\mathcal{F}(\theta(\bullet))$ under the training set $\mathcal{D}^{tr} = \left\{(x_n, y_n)\right\}_{n=1}^{N}$ at the learning stage. Here, $\theta(\bullet)$ represents the embedding model, and $\mathcal{F}$ is the base learner. At the inference stage, for an image x^*, the predicted label is obtained via $y^* = \mathcal{F}(\theta(x^*))$.

Few-shot learning aims to efficiently update the parameterized model so that the learned model can adapt to new tasks quickly. Few-shot learning approaches usually include two stages: the pre-training stage and the meta-testing stage. For the pre-training stage, we suppose that $\mathcal{D}^{tr} = \left\{(x_n, y_n)\right\}_{n=1}^{N}$ contains thousands of images for a large number of classes. For the meta-testing stage, we also suppose that $\mathcal{D}^{ts}$ and $\mathcal{M}^{ts}$ represent the meta-testing set and meta-testing tasks, respectively. Here, the $\mathcal{D}^{tr}$ and $\mathcal{D}^{ts}$ should be provided the different categories and $\mathcal{D}^{tr} \cap \mathcal{D}^{ts} = \emptyset$. The performance of few shot classification on meta-testing is adopted to evaluate the meta-learning approaches. We randomly choose a large number of tasks (or episodes) $\mathcal{M}^{ts} = \left\{\mathcal{M}_i = (T_i^{support}, T_i^{query})\right\}_{i=1}^{M}$, where M represents the number of tasks. For each task $\mathcal{M}_i$, we represent the support set as $T_i^{support} = \left\{(x_n, y_n) | n = 1, \ldots, K \times C, y_n \in C\right\}$. It contains C classes and K images per class. $T_i^{query} = \left\{(x_n, y_n) | n = 1, \ldots, Q \times C, y_n \in C\right\}$ denotes the query set with C classes and Q images per class.

4. Methodology

This section introduces the proposed hybrid representation base learner in detail. We first represent the base learner of the proposed method in Section 4.1. Then the hybrid representation base learner is elaborate in Section 4.2.

4.1. Base learner

Different few-shot learning approaches differ in the form of base learner $\mathcal{F}$ or embedding model $\theta(\cdot)$. In this paper, we mainly study the base learner $\mathcal{F}$. Concretely, we focus on introducing two popular few-shot learning approaches.

4.1.1. Nearest neighbor classifier based learner

Snell et al. [13] proposed the prototypical network. The prototypical network is a type of the nearest neighbor classifier-based learner, which can be considered a specific base learner approach since the representation of each class (i.e., the mean vector) does not require the participation of other classes. Given a query sample x^*, the base learner can be written as Eq. (1)

$$\mathcal{F} = \arg\max_{c} \frac{\exp(-d(\theta(x^*), \mu_c))}{\sum_{k=1}^{C} \exp(-d(\theta(x^*), \mu_k))} \tag{1}$$

where d is a distance metric, μ_c is the mean vectors of the embedded features in the support set with the c_{th} label.

4.1.2. Linear classifier base learner

Lee et al. [15] proposed to obtain the linear classifier base learner. The linear classifier base learner can be considered a shared base learner since it mainly concerns class differences. Given a query sample x^*, the base learner can be written as Eq. (2).

$$\mathcal{F} = \arg\max_{c} \frac{\exp(\theta(x^*) W_c)}{\sum_{k=1}^{C} \exp(\theta(x^*) W_k)} \tag{2}$$

where $W \in \mathbb{R}^{D \times C}$ is the classifier plane. D is the dimension of the embedding features, and C is the number of classes of the support set.

4.2. Hybrid representation base learner

This section is composed of three parts: (i) specific representation base learner; (ii) shared representation base learner; (iii) hybrid representation base learner. Fig. 2 shows the characteristics of the three approaches.

4.2.1. Specific representation base learner

The specific representation base learner assumes that the embedding features of the query set are separately represented in each feature subspace of the support set. Specifically, we use the support set $\theta(X^c)$ to fit the query set $\theta(X^*)$, where $\theta(X^c)$ represents the c_{th} class of support set. The objective function is defined as Eq. (3).

$$\arg\min_{S^1, \ldots, S^C} \frac{1}{C} \sum_{c=1}^{C} \left\{ \| S^c \theta(X^c) - \theta(X^*) \|_F^2 + \gamma_1 \| S^c \|_F^2 \right\} \tag{3}$$

where S^c represents the fitting coefficient of X^* with the c_{th} embedding features X^c. γ_1 is the regularization parameter to guarantee the closed form solution when the matrix $\theta(X^c)\theta(X^c)^T$ is not full rank. The Eq. (3) has the closed form solution as Eq. (4).

$$S^c = \left(\theta(X^*)\theta(X^c)^T\right) \times \left(\theta(X^c)\theta(X^c)^T + \gamma_1 U^c\right)^{-1} \tag{4}$$

In Eq. (4), the matrix $U^c \in \mathbb{R}^{K \times K}$ is the identity matrix and K represents the number of images in the c_{th} class of support set. When $S_i^c = [\frac{1}{K}, \ldots, \frac{1}{K}]$, the proposed specific representation base learner degenerates to the prototypical networks. Compared with prototypical networks, the proposed specific representation base learner can obtain more optimal descriptions in each class.

4.2.2. Shared representation base learner

Unlike specific representation base learner, shared representation base learner directly adopts all the support set embedding features to fit the query set embedding features. The objective function is as Eq. (5).

$$\arg\min_{S} \left\{ \| S\theta(X) - \theta(X^*) \|_F^2 + \gamma_2 \| S \|_F^2 \right\} \tag{5}$$

where γ_2 is the regularization parameter to guarantee the closed-form solution when the matrix $\theta(X)\theta(X)^T$ is not full rank. The Eq. (5) has the closed-form solution as Eq. (6).

$$S = \left(\theta(X^*)\theta(X)^T\right) \times \left(\theta(X)\theta(X)^T + \gamma_2 U\right)^{-1} \tag{6}$$

In Eq. (6), the matrix $U \in \mathbb{R}^{I \times I}$ is the identity matrix and I represents the number of images in the support set. The shared representation base learner can describe the query samples in the same subspace.

4.2.3. Hybrid representation base learner

In this section, we propose combining the shared representation base learner and the specific representation base learner to formulate the hybrid representation base learner. The objective function can be written as Eq. (7).

$$\begin{aligned} \arg\min_{\mathbf{S}} \alpha \| \varphi(\mathbf{X})\mathbf{S} - \varphi(\mathbf{Y}) \|_F^2 + \gamma \| \mathbf{S} \|_F^2 \\ + \tau \sum_{c=1}^{C} \| \varphi(\mathbf{X}^c)\mathbf{S}^c - \varphi(\mathbf{Y}^c) \|_F^2 \end{aligned} \tag{7}$$

where α and τ are the weights of the shared representation base learner and specific representation base learner, respectively. $\gamma = \alpha\gamma_2 + \tau\gamma_1$ is the regularization parameter. Let $[0, \ldots, \theta(X^c) \cdots, 0]$ be $\theta(\hat{X}^c)$. The solution of Eq. (7) can be easily solved as Eq. (8).

$$\begin{aligned} S = \left((\alpha + \tau)\theta(X^*)\theta(X)^T\right) \times \\ \left(\alpha\theta(X)\theta(X)^T + \gamma U + \tau \sum_{c=1}^{C} \theta(\hat{X}^c)\theta(\hat{X}^c)^T\right)^{-1} \end{aligned} \tag{8}$$

Given a query sample x_i^*, the base learner can be written as the maximization of probability assigned to class c using the softmax function as Eq. (9):

$$\mathcal{F} = \arg\max_{c} p(y^* = c | \theta(x_i^*)) \tag{9}$$

Here we define $p(y^* = c | \theta(x_i^*))$ as Eq. (10):

$$p(y^* = c | \theta(x_i^*)) = \frac{\exp\left(-\| \theta(x_i^*) - S_i^c \theta(X^c) \|\right)}{\sum_{k=1}^{C} \exp\left(-\| \theta(x_i^*) - S_i^k \theta(X^k) \|\right)} \tag{10}$$

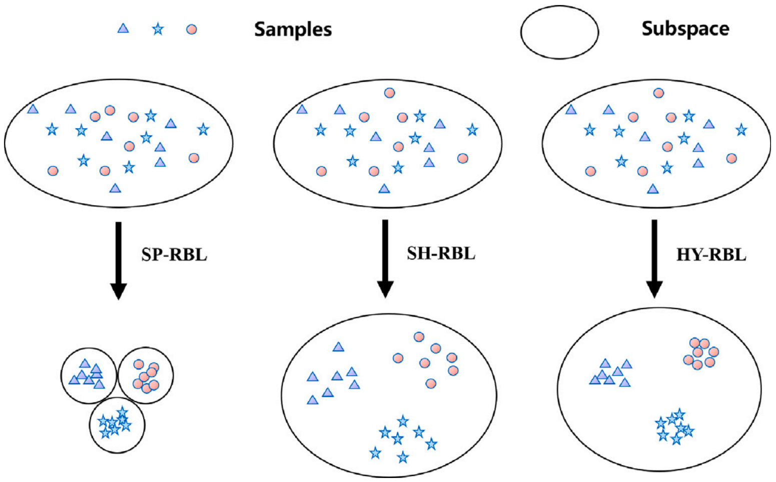


Fig. 2. Illustration of the variation of sample distribution under different base learners. Different circles represent different subspaces. The specific representation base learner represents the embedding feature of the query sample in each individual subspace (i.e., each specific class). However, overlaps or intersections among subspaces would lead to error discrimination for predicting the label of the query sample. The shared representation base learner represents the embedding feature of the query sample in the same subspace. However, it lacks the characteristics to describe each specific class, and the query sample's representation usually assigns a large weight to its neighbor support samples. The hybrid representation base learner represents the embedding feature of the query sample in the same subspace and holds the characteristics of each specific class.

Here, y^* is the predicted label of image sample x_i^*. $S_i = [S_i^1, \dots, S_i^c, \dots, S_i^C]$ represents the fitting coefficient vector of the sample $\theta(x_i^*)$ by the support set $\theta(X)$.

4.3. Hybrid representation base learner with self-training framework

We introduce a self-training framework to the HY-RBL, which is composed of three steps:

(i) We construct a hybrid representation base learner by Eq. (7). We obtain the fitting coefficient vector from Eq. (8) by support data X. Then predict the label of query data x^* by:

$$p(y^* = c|\theta(x_i^*)) = \frac{\exp(-\|\theta(x_i^*) - S_i^c\theta(X^c)\|)}{\sum_{k=1}^{C}\exp(-\|\theta(x_i^*) - S_i^k\theta(X^k)\|)} \tag{11}$$

where $p(y^* = c|\theta(x_i^*))$ denotes predicted query datas' soft-label matrices. Following, predict the (label) category of x^* by:

$$y_{new}^* = \arg\max_c p(y^* = c|\theta(x_i^*)) \tag{12}$$

(ii) Rank the values in soft-label matrices, then select the highest confidence query feature embedding $\theta(x_{new}^*)$, then asserting them corresponding label vectors y_{new}^*. We formulate this step as:

$$\begin{cases} \theta(X) = [\theta(X), \theta(x_{new}^*)] \\ \boldsymbol{Y} = [\boldsymbol{Y}, \boldsymbol{y}_{new}^*] \end{cases} \tag{13}$$

where $\boldsymbol{Y}$ and $\boldsymbol{y}_{new}^*$ denote the one-hot label matrices of support data.

(iii) Selecting one sample at a time and repeat **(i)**, **(ii)** until all the query data are selected.

5. Experiments

In this section, we mainly focus on showing and analyzing the experimental results conducted on four benchmark image datasets to demonstrate the superior performance to several state-of-art metric learning based few shot learning approaches.

5.1. Experimental settings—image datasets

We evaluate our proposed approaches on four benchmark few-shot image classification datasets: miniImageNet dataset [25], tieredImageNet dataset [25], CIFAR_FS dataset [26] and FC100 dataset [26].

The miniImageNet dataset is a standard benchmark dataset for the few-shot image classification task. It consists of 100 classes randomly split into 3 sections: 64 classes for meta-training, 16 classes for meta-validation, and 20 classes for meta-testing. The number of images for each class is 600, and the size of each image is 84 × 84.

The tieredImageNet dataset is larger than the miniImageNet dataset and has 608 classes grouped into 34 high-level categories. The number of images for each class is 600, and the size of each image is 84 × 84. The dataset is divided into 3 sections: 20 categories (351 classes) for meta-training, 6 categories (97 classes) for meta-validation, and 8 categories (160 classes) for meta-testing.

The CIFAR_FS dataset consists of 100 classes and is. The number of images for each class is 600, and the size of each image is 32 × 32. The dataset is divided into 3 sections: 64 classes for meta-training, 16 classes for meta-validation, and 20 classes for meta-testing.

The FC100 dataset has 100 classes grouped into 20 superclasses. The number of images for each class is 600, and the size of each image is 32 × 32. The dataset is divided into 3 sections: 12 superclasses (60 classes) for meta-training, 4 superclasses (20 classes) for meta-validation, and 4 superclasses (20 classes) for meta-testing.

5.2. Experimental settings—implementation details

For the feature embedding architecture, we adopt a ResNet-12 network and construct the self-supervision framework with rotation loss following [24]. It consists of 4 residual blocks (3 × 3 convolution layer, batch normalization layer, Leaky ReLU(0.1) layer), 4 2 × 2 max-pooling layers and 4 Dropout layers. After the last residual block, we apply a global average pooling and FC layer. The specific parameter settings are the same as [24].

We adopt stochastic gradient descent (SGD) optimizer with Nesterov momentum (0.9) for the optimizer. The dynamic learning rate is adopted during pre-training (The learning rate was initially set to 0.1, and then changed to 0.05, 0.025, and 0.0125 at epochs 30, 60, and 90, respectively). We set the batch size to 6 and the max epoch to 120. To avoid overfitting, we adopt the weight-decay strategy and set the parameter to 5×10^{-4}. Moreover, we adopt horizontal flips, random crop, and color-dithering data augmentation. The best model is chosen according to the classification precision testing on the meta-training set.

For meta-test, We set the parameter $\gamma = 1.7$, $\tau = 0.7$, $\alpha = 0.3$ for miniImageNet dataset and $\gamma = 0.9$, $\tau = 0.7$, $\alpha = 0.9$ for tieredImageNet dataset. And we set the γ to 1.3, τ to 0.3, α to 0.9 for CIFAR_FS dataset and γ to 0.9, τ to 0.1, α to 0.1 for FC100 dataset.

Table 1
The 5-way few-shot classification accuracies on miniImageNet with 95% confidence intervals over 600 episodes.

Method	Venue	Backbone	miniImageNet	
			1-shot	5-shot
FEAT [27]	CVPR,2020	CONV4	55.15 ± 0.20	71.61 ± 0.16
MELR [28]	ICLR,2021	CONV4	55.35 ± 0.43	72.27 ± 0.35
HY-RBL	–	CONV4	**57.30** ± 0.60	**72.62** ± 0.45
Fine-tuning [29]	ICLR,2020	WRN	65.73 ± 0.68	78.40 ± 0.52
S2M2$^{\star}_{E}$ [30]	WACV,2020	WRN	62.33 ± 0.25	79.35 ± 0.16
S2M2$^{\star}_{R}$ [30]	WACV,2020	WRN	64.93 ± 0.18	83.18 ± 0.11
AIM [31]	ICCV,2021	WRN	71.22 ± 0.57	82.25 ± 0.34
PSST [32]	CVPR,2021	WRN	64.16 ± 0.44	80.64 ± 0.32
HY-RBL	–	WRN	**72.29** ± 0.55	**84.35** ± 0.50
AFHN [33]	CVPR,2020	ResNet18	62.38 ± 0.72	78.16 ± 0.56
HY-RBL	–	ResNet18	**67.48** ± 0.92	**79.52** ± 0.61
DSN-MR [16]	CVPR,2020	ResNet12	64.60 ± 0.72	79.51 ± 0.50
ICI [12]	CVPR,2020	ResNet12	66.80	79.26
ODE [34]	CVPR,2021	ResNet12	67.76 ± 0.46	82.71 ± 0.31
CNL [35]	AAAI,2021	ResNet12	67.96 ± 0.98	83.36 ± 0.51
SSR [36]	NeurIPS,2021	ResNet12	68.10 ± 0.60	76.90 ± 0.40
MELR [28]	ICLR,2021	ResNet12	67.40 ± 0.43	83.40 ± 0.28
HY-RBL	–	ResNet12	**72.25** ± 0.81	**84.02** ± 0.58

Table 2
The 5-way few-shot classification accuracies on tieredImageNet with 95% confidence intervals over 600 episodes.

Method	Venue	Backbone	tieredImageNet	
			1-shot	5-shot
MELR [28]	ICLR,2021	CONV4	56.38 ± 0.48	73.22 ± 0.41
HY-RBL	–	CONV4	**68.30** ± 0.75	**79.13** ± 0.52
Fine-tuning [29]	ICLR,2020	WRN	73.34 ± 0.71	85.50 ± 0.50
S2M2$^{\star}_{R}$ [30]	WACV,2020	WRN	73.71 ± 0.22	88.59 ± 0.14
HY-RBL	–	WRN	**81.80** ± 0.80	**90.03** ± 0.58
CTM [37]	CVPR,2019	ResNet18	64.78 ± 0.11	81.05 ± 0.52
HY-RBL	–	ResNet18	**80.25** ± 0.91	**88.36** ± 0.65
DSN-MR [16]	CVPR,2020	ResNet12	67.39 ± 0.82	82.85 ± 0.56
ICI [12]	CVPR,2020	ResNet12	80.79	87.92
ODE [34]	CVPR,2021	ResNet12	71.89 ± 0.52	85.96 ± 0.35
CNL [35]	AAAI,2021	ResNet12	73.42 ± 0.95	87.72 ± 0.75
SSR [36]	NeurIPS,2021	ResNet12	81.20 ± 0.60	85.70 ± 0.40
MELR [28]	ICLR,2021	ResNet12	72.14 ± 0.51	87.01 ± 0.35
HY-RBL	–	ResNet12	**81.98** ± 0.80	**89.85** ± 0.57

Table 3
The 5-way few-shot classification accuracies on CIFAR-FS with 95% confidence intervals over 600 episodes.

Method	Venue	Backbone	CIFAR-FS	
			1-shot	5-shot
ProtoNet [15]	CVPR,2019	CONV4	55.50 ± 0.70	72.00 ± 0.60
MAML [15]	CVPR,2019	CONV4	58.90 ± 1.90	71.50 ± 1.00
HY-RBL	–	CONV4	**62.12** ± 0.92	**74.31** ± 0.61
Fine-tuning [29]	ICLR,2020	WRN	76.58 ± 0.68	85.79 ± 0.50
S2M2$^{\star}_{R}$ [30]	WACV,2020	WRN	74.81 ± 0.19	87.47 ± 0.13
S2M2$^{\star}_{E}$ [30]	WACV,2020	WRN	72.63 ± 0.16	86.12 ± 0.26
HY-RBL	–	WRN	**78.94** ± 0.82	**87.77** ± 0.59
S2M2$^{\star}_{R}$ [30]	WACV,2020	ResNet18	63.66 ± 0.17	76.07 ± 0.19
S2M2$^{\star}_{E}$ [30]	WACV,2020	ResNet18	61.95 ± 0.11	75.09 ± 0.16
HY-RBL	–	ResNet-18	**80.22** ± 0.90	**88.75** ± 0.67
DSN-MR [16]	CVPR,2020	ResNet12	75.60 ± 0.90	86.20 ± 0.60
SSR [36]	NeurIPS,2021	ResNet12	76.80 ± 0.60	83.70 ± 0.40
TDE [24]	Neurocomputing,2022	ResNet12	78.30 ± 1.13	87.17 ± 0.67
HY-RBL	–	ResNet12	**79.66** ± 0.88	**88.04** ± 0.65

Table 4
The 5-way few-shot classification accuracies on FC100 with 95% confidence intervals over 600 episodes.

Method	Venue	Backbone	FC100	
			1-shot	5-shot
ProtoNet [15]	CVPR,2019	CONV4	35.30 ± 0.60	48.60 ± 0.60
HY-RBL	–	CONV4	**38.23** ± 0.92	**51.66** ± 0.80
Fine-tuning [29]	ICLR,2020	WRN	43.16 ± 0.59	57.57 ± 0.55
HY-RBL	–	WRN	**44.05** ± 0.89	**58.34** ± 0.75
HY-RBL	–	ResNet-18	**44.46** ± 0.95	**58.65** ± 0.81
TADAM [26]	NeurIPS,2018	ResNet12	40.10 ± 0.40	56.10 ± 0.40
DenseCls [38]	CVPR,2019	ResNet12	42.04 ± 0.17	57.05 ± 0.16
MetaOpt [15]	CVPR,2019	ResNet12	41.10 ± 0.60	55.50 ± 0.60
MABAS [39]	ECCV,2020	ResNet12	41.74 ± 0.73	57.11 ± 0.75
HY-RBL	–	ResNet12	**45.42** ± 0.83	**60.53** ± 0.79

5.3. Experimental results

We conduct plenty of experiments on four few-shot learning datasets using different backbones (such as CONV4 [28], ResNet12 [24], ResNet18 [40] and WRN [30]). We list the experimental results in Table 1, 2, 3 and 4. For fairness, we compare our proposed HY-RBL with several state-of-the-art methods under the same backbone. We obtain the few-shot classification accuracies on all datasets with 95% confidence intervals over 600 episodes. The top two results with different backbones are shown in *underline* and *bold*.

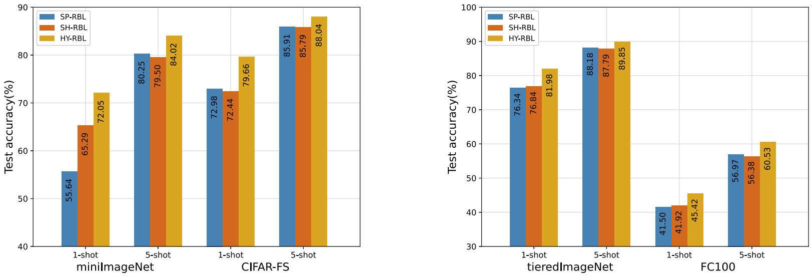
The performance of our proposed HY-RBL is better than other methods to varying degrees on different datasets and backbones. (1) To be more specific, in miniImageNet, our method exceeds others by at least 1.95% on 5-way 1-shot case, 0.35% on 5-way 5-shot case with the CONV4 backbone; in tieredImageNet, our method exceeds others by at least 11.92% on 5-way 1-shot case, 5.91% on 5-way 5-shot case with the CONV4 backbone; in CIFAR-FS, our method exceeds others by at least 3.22% on 5-way 1-shot case, 2.31% on 5-way 5-shot case with the CONV4 backbone; in FC100, our method exceeds others by at least 2.93% on 5-way 1-shot case, 3.06% on 5-way 5-shot case with the CONV4 backbone.

(2) in miniImageNet, our method exceeds others by at least 1.07% and 1.17% on 5-way 1-shot case and 5-way 5-shot case with the WRN backbone, respectively; in tieredImageNet, our method exceeds others by at least 8.09% and 1.44% on 5-way 1-shot case and 5-way 5-shot case with the WRN backbone, respectively; in CIFAR-FS, our method exceeds others by at least 2.36% and 0.30% on 5-way 1-shot case and 5-way 5-shot case with the WRN backbone, respectively; in FC100, our method exceeds others by at least 0.89% and 0.77% on 5-way 1-shot case and 5-way 5-shot case with the WRN backbone, respectively;

(3) our method exceeds others by at least 5.10% on 5-way 1-shot case, 1.36% on 5-way 5-shot case with the ResNet18 backbone in miniImageNet; our method exceeds others by at least 5.47% on 5-way 1-shot case, 7.31% on 5-way 5-shot case with ResNet18 backbone in tieredImageNet; our method exceeds others by at least 16.56% on 5-way 1-shot case, 12.68% on 5-way 5-shot case with the ResNet18 backbone in CIFAR-FS;

(4) our method exceeds others by at least 0.62% on 5-way 5-shot case with ResNet12 backbone in the miniImageNet; our method exceeds others by at least 0.78% on 5-way 1-shot case, 1.93% on 5-way 5-shot case with ResNet12 backbone in the tieredImageNet; our method exceeds others by at least 1.36% on 5-way 1-shot case, 0.87% on 5-way 5-shot case with ResNet12 backbone in the CIFAR-FS; our method exceeds others by at least 3.38% on 5-way 1-shot case, 3.42% on 5-way 5-shot case with ResNet12 backbone in the FC100.

(5) In few-shot learning tasks, four frequently used feature extractors are CONV4, ResNet12, ResNet18, and WRN. After comparing the usage of these extractors on the same dataset, we discovered that



(a) miniImageNet and CIFAR-FS

(b) tieredImageNet and FC100

Fig. 3. Comparison results among three RBL methods.

Table 5
The computation cost comparison. All the complexity is the 5way-1shot case; the number of the query is 15.

Method	Computation cost
MetaOpt	$T\times C\times O(n)$
ProtoNet	$T\times C\times O(n)$
MAML	$T\times C\times O(n)$
DSN-MR	$O[T\times C\times (n\times K)+T\times C\times K]$
HY-RBL	$O([(T\times C+N\times C)\times n+(N\times C\times)(N\times C)]\times(N\times C))$

CONV4 produced the lowest while ResNet12 and WRN achieved the highest results. Conv4, composed of only four convolutional layers, leads to weaker feature extraction capabilities and poorer classification performance than other extractors. On the other hand, ResNet12 and WRN have more robust feature extraction capabilities due to their deep network structures, rendering them advantageous for subsequent classification tasks after sufficient training.

(6) Our proposed Hybrid Representation Base Learner (HY-RBL) method performs better than the MetaOpt, a shared representation base learner, and the Prototypical Network, a specific representation base learner introduced in Section 4.1. HY-RBL combines both the specific and shared representation base learners to leverage the advantages of both classifiers, leading to superior performance in classification tasks. The specific representation base learner learns distinctive features of each class, while the shared representation base learner learns common features across different classes. Combining these two methods makes the classification process more accurate.

(7) The proposed hybrid representation based shared representation base learner and specific representation base learner is compared to other metric learning methods, such as DSN-MR [16], ICI [12], TDE [24], SSR [36]. The results show that HY-RBL outperforms these methods in the ability to measure sample categories. In mini-ImageNet, our method outperforms others at least 4.15% on 5-way 1-shot case, 4.51% on 5-way 5-shot case; in tiered-ImageNet, our method outperforms others at least 0.78% on 5-way 1-shot case, 1.93% on 5-way 5-shot case. in CIFAR-FS, our method outperforms others at least 1.36% on 5-way 1-shot case, 0.87% on 5-way 5-shot case. in FC100, our method outperform others at least 4.32% on 5-way 1-shot case, 5.03% on 5-way 5-shot case.

5.4. Complexity of computation

Table 5 shows the computational complexity of our HY-RBL approach, which is $O([(T\times C+N\times C)\times n+(N\times C\times(N\times C))]\times(N\times C))$, where T and N are the numbers of query and support samples, n denotes the dimension of the data feature and C represents the number of categories. Compared to the complexity of other methods, our method is somewhat slower because our method is non-parametric, and the support set and query set are both involved in the calculation in the classification stage.

5.5. Ablation studies

5.5.1. Comparison among three RBL methods

As shown in Fig. 3, SP-RBL performs better than CHS-RBL on miniImageNet, tieredImageNet, and FC100 datasets on the 5-way 1-shot case, while it comes to the opposite except on the 5-way 5-shot case. Fortunately, HY-RBL combines the two methods above and achieves better precision than others with ResNet12 backbone on all datasets on two kinds of experiments.

5.5.2. t-SNE visualization

We reduce the dimensionality using t-SNE to show the distribution of the feature extracted from the query set in the miniImageNet dataset via these three approaches. As shown in Fig. 4, for the SP-RBL method, the blue and purple categories are intertwined. For the SH-RBL method, there is also a certain interweaving among classes. The HY-RBL method shows a clear distinction among the categories.

5.5.3. Influence of self-training framework

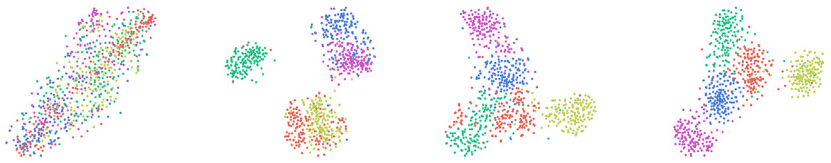
We propose a self-training few-shot learning method and expand the self-training framework to our approach using query data. Fig. 5 demonstrates the effectiveness of the self-training framework on four datasets. On all four datasets, the performance of HY-RBL with the self-training framework improved to varying degrees, especially in the 5-way 1-shot case, which improved **11.62**% on the miniImageNet.

5.5.4. Influences of meta-testing shot

In the meta-test phase, the amount of support data is an essential factor affecting classification performance. We further conducted experiments on 5-way 2-shot, 10-shot, 15-shot, and 20-shot cases. As shown in Fig. 6, the classification performance increases gradually with the increase of shots, especially in 2-shot cases. While performance slowly saturates on the 20-shot case.

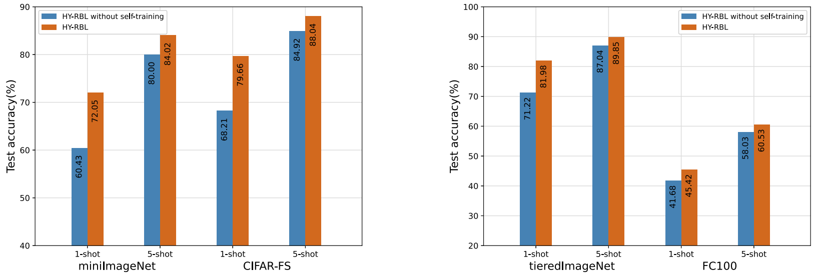
5.5.5. Influence of parameters

In the meta-test stage, the proposed HY-RBL base learner requires manual adjustment of three key parameters: α, τ, and γ. To investigate the impact of different parameter selections on the final classification results, we conducted experiments on the miniImageNet and tieredImageNet datasets. Fig. 7 presents the experimental results under different parameters. Notably, we found that the HY-RBL base learner is insensitive to changes in parameter γ, and we obtained similar final classification results for different parameter settings. As for parameter τ, a gradual increase in the parameter leads to a gradual decrease in the experimental results in the miniImageNet dataset. However, this trend is prolonged, and the range of accuracy change is relatively low. Contrastingly, in the tieredImageNet dataset, the results show an initial increase, reaching a maximum at 0.7 and then gradually decreasing. For parameter α, the experimental results' overall change range is small, with a gradually decreasing trend as the parameter gradually increases in the miniImageNet dataset. As for the tieredImageNet



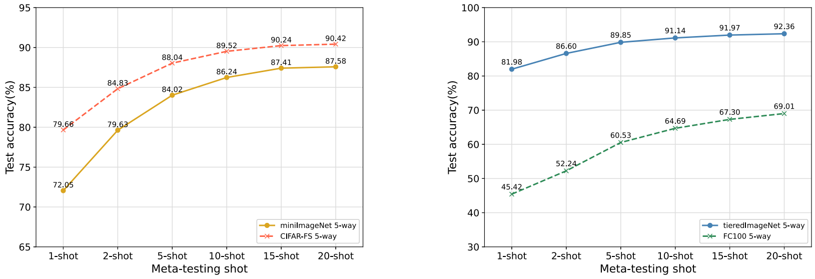
(a) Original image (b) SP-RBL (c) SH-RBL (d) HY-RBL

Fig. 4. The t-SNE visualization of query set in a 5-way problem under three different approaches.



(a) miniImageNet and CIFAR-FS (b) tieredImageNet and FC100

Fig. 5. Influences of Self-training Framework.



(a) miniImageNet and CIFAR-FS (b) tieredImageNet and FC100

Fig. 6. Comparison results with different meta-testing shot.

dataset, the experimental results remained almost unchanged despite the parameter changes, indicating that the model is not sensitive to this parameter.

5.5.6. *Performance under domain shift*

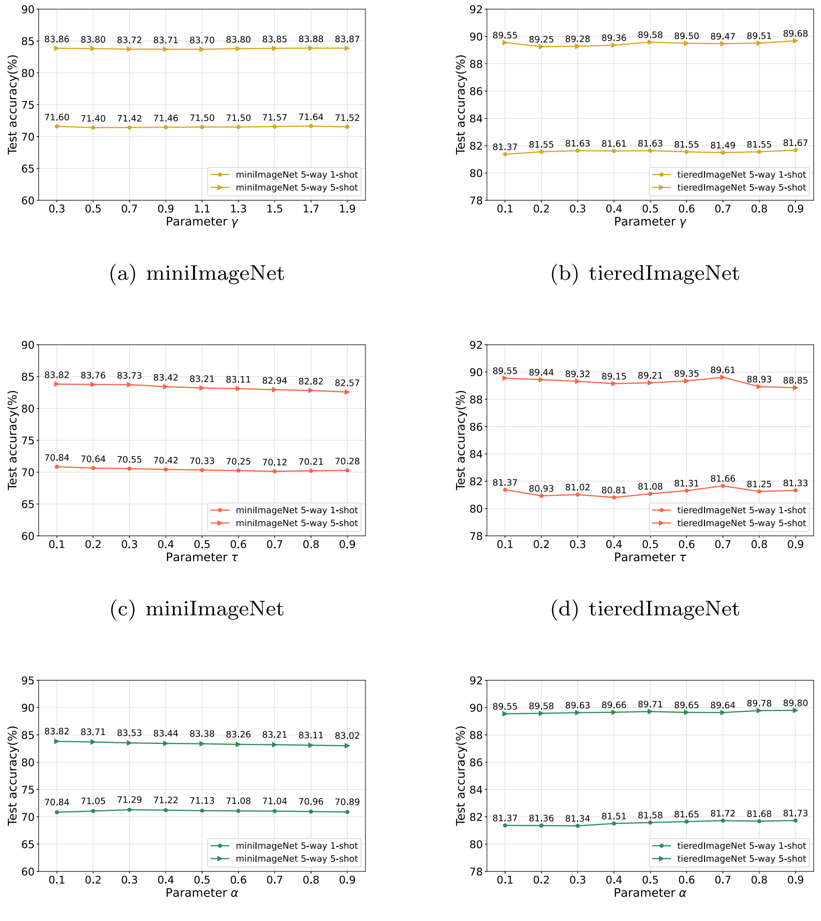
In this section, we evaluate the performance of our proposed class hybrid representation base learning with ResNet12 backbone under domain shift. We adopt the meta-training models on the miniImageNet dataset and evaluate them on the CUB-200-2011 dataset [41]. Following the evaluation protocol of [40], we randomly split the dataset into 100 base, 50 validation, and 50 novel classes. We use 50 validation and 50 novel classes. Table 6 compares our proposed HY-RBL with several typical methods. Experimental results show that our method has excellent domain migration performance.

Table 6
Details of four benchmark few-shot image classification dataset.

Method	mini-ImageNet ⟶ CUB	
	5-way 1-shot	5-way 5-shot
TIM-GD [42]	–	71.00
MatchNet [43]	51.65	69.14
ProtoNet [13]	50.01	72.02
MetaOpt [15]	44.79	64.98
KNN [44]	50.84	71.25
S2M2 [30]	48.24	70.44
TDE-FSL†	**58.32**	**76.15**

6. Conclusion

Designing a base learner is significant for few-shot image classification tasks. This paper examines the representation-residual base learner and introduces two types of learners: specific representation-base learner and shared representation-base learner. capable of completing the few-shot image classification task. Both are capable of completing the task of few-shot image classification. Moreover, we introduce a novel hybrid representation base learner which combines the benefits of both specific and shared representation base learner. This hybrid learner can effectively prevent the over-summarization of training samples and generalize better in the meta-test phase. The specific and shared representation-based learners are integrated in a unique way to create a more robust learning approach that is geared towards achieving improved performance. The proposed methods have achieved competitive performance with recent state-of-the-art few-shot learning approaches by conducting experiments on several benchmark datasets. The proposed HY-RBL presents some limitations that require further examination: (1) In the meta-test stage, the proposed HY-RBL has three parameters that need to be adjusted manually, and the



(a) miniImageNet (b) tieredImageNet

(c) miniImageNet (d) tieredImageNet

(e) miniImageNet (f) tieredImageNet

Fig. 7. Comparison results with different parameters on miniImageNet and TieredImageNet dataset.

optimal parameters of different datasets are different, which limits the usability of the method. (2) The proposed HY-RBL has a higher time complexity. In the future, we plan to adopt the meta-learning strategy to expand and improve the proposed method, and we will also introduce HY-RBL to train the feature extractor in the meta-training stage. Moreover, we will explore nonlinear base learners for future work, such as kernel methods.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

The paper was supported by the National Natural Science Foundation of China (No. 62372468), the Major Basic Research Projects in Shandong Province (No. ZR2023ZD32), the Shandong Natural Science Foundation (No. ZR2023MF008), the Qingdao Natural Science Foundation (No. 23-2-1-161-zyyd-jch), the China Postdoctoral Science Foundation (No. 2023M743266), the Zhejiang Provincial Postdoctoral Excellence Program (No. ZJ2023067), the Science and Technology Development Fund, Macau SAR (No. 0049/2022/A1), the University of Macau (No. MYRG2022-00072-FST).

References

[1] S. Miao, S. Du, R. Feng, Y. Zhang, H. Li, T. Liu, L. Zheng, W. Fan, Balanced single-shot object detection using cross-context attention-guided network, Pattern Recognit. 122 (2022) 108258.

[2] H. Chen, C. Li, G. Wang, X. Li, M.M. Rahaman, H. Sun, W. Hu, Y. Li, W. Liu, C. Sun, et al., Gashis-transformer: a multi-scale visual transformer approach for gastric histopathological image detection, Pattern Recognit. 130 (2022) 108827.

[3] Y. Zhou, X. Li, Y. Zhou, Y. Wang, Q. Hu, W. Wang, Deep collaborative multi-task network: A human decision process inspired model for hierarchical image classification, Pattern Recognit. 124 (2022) 108449.

[4] H. Chen, C. Li, X. Li, M.M. Rahaman, W. Hu, Y. Li, W. Liu, C. Sun, H. Sun, X. Huang, et al., Il-mcam: an interactive learning and multi-channel attention mechanism-based weakly supervised colorectal histopathology image classification approach, Computers in Biology and Medicine 143 (2022) 105265.

[5] S. Yi, H. Ma, X. Wang, T. Hu, X. Li, Y. Wang, Weakly-supervised semantic segmentation with superpixel guided local and global consistency, Pattern Recognit. 124 (2022) 108504.

[6] S. Shao, L. Xing, R. Xu, W. Liu, Y.-J. Wang, B.-D. Liu, Mdfm: multi-decision fusing model for few-shot learning, IEEE Trans. Circuits Syst. Video Technol. (2021).

[7] H. Huang, Z. Wu, W. Li, J. Huo, Y. Gao, Local descriptor-based multi-prototype network for few-shot learning, Pattern Recognit. 116 (2021) 107935.

[8] S. Shao, Y. Wang, B. Liu, W. Liu, Y. Wang, B. Liu, Fads: fourier-augmentation based data-shunting for few-shot classification, IEEE Trans. Circuits Syst. Video Technol. (2023).

[9] H. Xu, J. Wang, H. Li, D. Ouyang, J. Shao, Unsupervised meta-learning for few-shot learning, Pattern Recognit. 116 (2021) 107951.

[10] S. Shao, Y. Bai, Y. Wang, B. Liu, Y. Zhou, DeIL: direct-and-inverse CLIP for open-world few-shot learning, in: Computer Vision and Pattern Recognition, 2024.

[11] S. Shao, Y. Bai, Y. Wang, B. Liu, B. Liu, Collaborative consortium of foundation models for open-world few-shot learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 38, No. 5, 2024, pp. 4740–4747.

[12] Y. Wang, C. Xu, C. Liu, L. Zhang, Y. Fu, Instance credibility inference for few-shot learning, in: Computer Vision and Pattern Recognition, 2020, pp. 12836–12845.

[13] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, in: Neural Information Processing Systems, 2017, pp. 4077–4087.

[14] L. Bertinetto, J.F. Henriques, P. Torr, A. Vedaldi, Meta-learning with differentiable closed-form solvers, in: Proceedings of the International Conference on Learning Representations, 2019.

[15] K. Lee, S. Maji, A. Ravichandran, S. Soatto, Meta-learning with differentiable convex optimization, in: Computer Vision and Pattern Recognition, 2019, pp. 10657–10665.

[16] C. Simon, P. Koniusz, R. Nock, M. Harandi, Adaptive subspaces for few-shot learning, in: Computer Vision and Pattern Recognition, 2020, pp. 4136–4145.

[17] S. Shao, L. Xing, Y. Wang, R. Xu, C. Zhao, Y.-J. Wang, B.-D. Liu, MHFC: Multi-head feature collaboration for few-shot learning, in: ACM International Conference on Multimedia, 2021, pp. 4193–4201.

[18] C. Finn, P. Abbeel, S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in: International Conference on Machine Learning, 2017, pp. 1126–1135.

[19] A. Nichol, J. Achiam, J. Schulman, On first-order meta-learning algorithms, 2018, arXiv preprint arXiv:1803.02999.

[20] X. Song, W. Gao, Y. Yang, K. Choromanski, A. Pacchiano, Y. Tang, ES-MAML: Simple hessian-free meta learning, in: International Conference on Learning Representations, 2020.

[21] N. Mishra, M. Rohaninejad, X. Chen, P. Abbeel, A simple neural attentive meta-learner, in: Proceedings of the International Conference on Learning Representations, 2018.

[22] S. Ravi, H. Larochelle, Optimization as a model for few-shot learning, in: Proceedings of the International Conference on Learning Representations, 2017.

[23] C. Zhang, Y. Cai, G. Lin, C. Shen, Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers, in: Computer Vision and Pattern Recognition, 2020, pp. 12203–12213.

[24] L. Xing, S. Shao, W. Liu, A. Han, X. Pan, B.-D. Liu, Learning task-specific discriminative embeddings for few-shot image classification, Neurocomputing (2022).

[25] M. Ren, E. Triantafillou, S. Ravi, J. Snell, K. Swersky, J.B. Tenenbaum, H. Larochelle, R.S. Zemel, Meta-learning for semi-supervised few-shot classification, in: International Conference on Learning Representations, 2018.

[26] B. Oreshkin, P.R. López, A. Lacoste, Tadam: Task dependent adaptive metric for improved few-shot learning, in: Neural Information Processing Systems, 2018, pp. 721–731.

[27] H.-J. Ye, H. Hu, D.-C. Zhan, F. Sha, Few-shot learning via embedding adaptation with set-to-set functions, in: Computer Vision and Pattern Recognition, 2020, pp. 8808–8817.

[28] N. Fei, Z. Lu, T. Xiang, S. Huang, Melr: Meta-learning via modeling episode-level relationships for few-shot learning, in: International Conference on Learning Representations, 2021.

[29] G.S. Dhillon, P. Chaudhari, A. Ravichandran, S. Soatto, A baseline for few-shot image classification, in: International Conference on Learning Representations, 2020.

[30] P. Mangla, N. Kumari, A. Sinha, M. Singh, B. Krishnamurthy, V.N. Balasubramanian, Charting the right manifold: Manifold mixup for few-shot learning, in: IEEE Winter Conference on Applications of Computer Vision, 2020, pp. 2218–2227.

[31] E. Lee, C.-H. Huang, C.-Y. Lee, Few-shot and continual learning with attentive independent mechanisms, in: International Conference on Computer Vision, 2021, pp. 9455–9464.

[32] Z. Chen, J. Ge, H. Zhan, S. Huang, D. Wang, Pareto self-supervised training for few-shot learning, in: Conference on Computer Vision and Pattern Recognition, 2021, pp. 13663–13672.

[33] K. Li, Y. Zhang, K. Li, Y. Fu, Adversarial feature hallucination networks for few-shot learning, in: Computer Vision and Pattern Recognition, 2020, pp. 13470–13479.

[34] C. Xu, C. Liu, L. Zhang, C. Wang, J. Li, F. Huang, X. Xue, Y. Fu, Learning dynamic alignment via meta-filter for few-shot learning, in: Computer Vision and Pattern Recognition, 2021, pp. 5182–5191.

[35] J. Zhao, Y. Yang, X. Lin, J. Yang, L. He, Looking wider for better adaptive representation in few-shot learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 12, 2021, pp. 10981–10989.

[36] X. Shen, Y. Xiao, S.X. Hu, O. Sbai, M. Aubry, Re-ranking for image retrieval and transductive few-shot classification, Neural Inf. Process. Syst. 34 (2021) 25932–25943.

[37] H. Li, D. Eigen, S. Dodge, M. Zeiler, X. Wang, Finding task-relevant features for few-shot learning by category traversal, in: Computer Vision and Pattern Recognition, 2019, pp. 1–10.

[38] Y. Lifchitz, Y. Avrithis, S. Picard, A. Bursuc, Dense classification and implanting for few-shot learning, in: Computer Vision and Pattern Recognition, 2019, pp. 9258–9267.

[39] J. Kim, H. Kim, G. Kim, Model-agnostic boundary-adversarial sampling for test-time generalization in few-shot learning, in: European Conference on Computer Vision, 2020, pp. 599–617.

[40] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C.F. Wang, J.-B. Huang, A closer look at few-shot classification, in: International Conference on Learning Representations, 2019.

[41] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, The caltech-ucsd birds-200–2011 dataset, 2011, Computation & Neural Systems Technical Report.

[42] M. Boudiaf, Z.I. Masud, J. Rony, J. Dolz, P. Piantanida, I.B. Ayed, Transductive information maximization for few-shot learning, in: Neural Information Processing Systems, 2020, pp. 2445–2457.

[43] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al., Matching networks for one shot learning, in: Neural Information Processing Systems, vol. 29, 2016, pp. 3630–3638.

[44] W. Li, L. Wang, J. Xu, J. Huo, Y. Gao, J. Luo, Revisiting local descriptor based image-to-class measure for few-shot learning, in: Computer Vision and Pattern Recognition, 2019, pp. 7260–7268.

Bao-Di Liu was born in Shandong, China. He received his Bachelor degree and Master degree in China University of Petroleum in 2004 and 2007 respectively. Currently, he is a Ph. D. student from the Department of Electronic Engineering, Tsinghua University, China. His research interests include computer vision and machine learning.

Shuai Shao received his M.S. degree and Ph.D. degree in College of Control Science and Engineering, China University of Petroleum (East China). Currently, he is pursuing his postdoctoral research at Zhejiang Lab.

Chunyan Zhao received her B.S. degree in Shandong University. Her main research interests include machine learning and computer vision.

Lei Xing received his B.S. degree in College of Oceanography and Space Informatics, China University of Petroleum (East China). Currently, he is pursuing his M.S. degree in College of Oceanography and Space Informatics, China University of Petroleum (East China). His main research interests include machine learning and computer vision.

Weifeng Liu is currently a Professor with the College of Control Science and Engineering, China University of Petroleum (East China), China. He received the double B.S. degree in automation and business administration and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2002 and 2007, respectively. He was a Visiting Scholar with the Centre for Quantum Computation & Intelligent Systems, Faculty of Engineering & Information Technology, University of Technology Sydney, Sydney, Australia, from 2011 to 2012. His current research interests include pattern recognition and machine learning. He has authored or co-authored a dozen papers in top journals and prestigious conferences including 10 ESI Highly Cited Papers and 3 ESI Hot Papers. Dr. Weifeng Liu serves as associate editor for Neural Processing Letter, co-chair for IEEE SMC technical committee on cognitive computing, and guest editor of special issue for Signal Processing, IET Computer Vision, Neurocomputing, and Remote Sensing. He also serves dozens of journals and conferences.

Weijia Cao received her Master's and Ph.D. degrees in computer science at University of Macau, Macao, China, in 2013 and 2017, respectively. She is currently an assistant researcher at Aerospace Information Research Institute, Chinese Academy of Sciences. Her main research interests revolve around multimedia encryption, machine learning, and remote sensing image processing.

Dr. Zhou is a Fellow of the International Society for Optical Engineering (SPIE), and a Senior Member of the IEEE and CCF (China Computer Federation). He was a recipient of the Third Price of Macao Natural Science Award in 2014 and 2020. He is a Co-Chair of Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society. He serves as an Associate Editor for IEEE Transactions on Neutral Networks and Learning Systems (TNNLS), IEEE Transactions on Cybernetics (TCYB), IEEE Transactions on Circuits and Systems for Video Technology (TCSVT), IEEE Transactions on Geoscience and Remote Sensing (TGRS), and four other journals. He was listed as "World's Top 2% Scientists" on the Stanford University Releases List in 2020, 2021 and the "Highly Cited Researcher" in the Web of Science in 2020, 2021.