Deep metric learning for few-shot image classification: A Review of recent developments

Xiaoxu Li[a,b,1], Xiaochen Yang[c,1], Zhanyu Ma[b,*], Jing-Hao Xue[d]

[a] *School of Computer and Communication, Lanzhou University of Technology, Lanzhou, 730050, China*
[b] *Pattern Recognition and Intelligent System Laboratory, School of Artifical Intelligence, Beijing University of Posts and Telecommunications, Beijing, 100876, China*
[c] *School of Mathematics and Statistics, University of Glasgow, Glasgow, G12 8QQ, UK*
[d] *Department of Statistical Science, University College London, London, WC1E 6BT, UK*

Abstract

Few-shot image classification is a challenging problem that aims to achieve the human level of recognition based only on a small number of training images. One main solution to few-shot image classification is deep metric learning. These methods, by classifying unseen samples according to their distances to few seen samples in an embedding space learned by powerful deep neural networks, can avoid overfitting to few training images in few-shot image classification and have achieved the state-of-the-art performance. In this paper, we provide an up-to-date review of deep metric learning methods for few-shot image classification from 2018 to 2022 and categorize them into three groups according to three stages of metric learning, namely learning feature embeddings, learning class representations, and learning distance measures. Under this taxonomy, we identify the trends of transitioning from

[*] Corresponding author
Email address: `mazhanyu@bupt.edu.cn.` (Zhanyu Ma)
[1] X. Li and X. Yang contribute equally.

learning task-agnostic features to task-specific features, from simple computation of prototypes to computing task-dependent prototypes or learning prototypes, from using analytical distance or similarity measures to learning similarities through convolutional or graph neural networks. Finally, we discuss the current challenges and future directions of few-shot deep metric learning from the perspectives of effectiveness, optimization and applicability, and summarize their applications to real-world computer vision tasks.
Keywords: Few-shot learning, Metric learning, Image classification, Deep neural networks

1. Introduction

Image classification is an important task in machine learning and computer vision. With the rapid development of deep learning, recent years have witnessed breakthroughs in this area [1, 2, 3, 4]. Such progress, however, hinges on collecting and labeling a vast amount of data (in the order of millions), which can be difficult and costly. More severely, this learning mechanism is in stark contrast with that of humans, where one or few examples suffice for learning a new concept [5]. Therefore, to reduce the data requirement and imitate human intelligence, many researchers started to focus on few-shot classification [6, 7, 8], i.e., learning a classification rule from few (typically 1-5) labeled examples.

The biggest challenge in few-shot classification is a high risk of model overfitting to the few labeled training samples. To alleviate this problem, researchers have proposed various approaches, such as meta-learning methods, transfer learning methods, and metric learning methods. Meta-learning methods train a meta-learner on many different classification tasks to extract generalizable knowledge, which enables rapid learning on a new related task with few examples [7, 9]. Transfer learning methods presume shared knowledge between the source and target domains, and fine-tune the model trained on abundant source data to fit few labeled target samples [10, 11]. Metric learning methods learn feature embeddings [6] and/or distance measures (or inversely, similarity measures) [12] and classify an unseen sample based on its distance to labeled samples or class representations; samples of the same class are expected to locate close together in the embedding space and samples of different classes should be far apart. Note that the above methods can be applied simultaneously, for example learning feature embeddings of metric learning methods by using a meta-learning strategy [7].

In this paper, we present a review of recent deep metric learning methods for few-shot image classification. Metric learning methods deserve special attention as they do not require learning additional parameters for new classes once the metric is learned, and thus able to avoid overfitting to the few labeled samples of new classes in few-shot learning. They have also demonstrated impressive classification performance on benchmark datasets. Moreover, in this review we decouple metric learning into three learning stages, namely learning feature embeddings, learning class representations, and learning distance measures. Such decomposition facilitates exchange of ideas between researchers from two underpinning communities: few-shot image classification and deep metric learning. For example, latest developments in learning generalizable feature embeddings can be adopted for few-shot image classification, and the idea of learning prototypes, one type of class representations, can be extended for long-tailed visual recognition [13].

A number of surveys on few-shot learning (FSL) have been published or preprinted. [14] is the first survey on small sample learning, summarizing methods for different small sample learning scenarios, including zero-shot learning and FSL, and for various tasks, such as image classification, object detection, visual question answering, and neural machine translation. Since the survey was conducted early in 2018, it includes relatively limited work on few-shot classification, particularly metric learning methods. [15] provides the first comprehensive review on FSL. In addition to defining FSL and distinguishing it from related machine learning problems, the authors discuss FSL from the fundamental perspective of error decomposition in supervised learning and classify all methods in terms of augmenting the training data for reducing the estimation error, learning models from prior knowledge for constraining the hypothesis space and reducing the approximation error, and learning initializations or optimizers which improve the search for the optimal hypothesis within the hypothesis space. The survey has limited coverage on metric learning methods and categorize them all under learning embedding models, which does not fully describe the merits of these methods. [16] is another comprehensive survey, reviewing literature over a long period from the 2000s to 2020 as well as summarizing applications of FSL in various fields. It includes early, non-deep approaches of metric learning methods and, since the survey emphasizes on meta-learning methods, categorizes most recent, deep approaches under meta-learning as learning-to-measure. Compared with [16] which links different meta-learning metric learning methods to three classical methods, our review provides a deeper insight into how metric learning

Conferences	Journals
AAAI Conference on Artificial Intelligence (AAAI)	IEEE Trans. on Circuits and Systems for Video Technology (TCSVT)
Int. Conference on Artificial Intelligence and Statistics (AISTATS)	IEEE Trans. on Image Processing (TIP)
Conference on Computer Vision and Pattern Recognition (CVPR)	IEEE Trans. on Multimedia (TMM)
European Conference on Computer Vision (ECCV)	IEEE Trans. on Neural Networks and Learning Systems (TNNLS)
Int. Conference on Computer Vision (ICCV)	IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)
Int. Conference on Learning Representations (ICLR)	Pattern Recognition (PR)
Int. Joint Conference on Artificial Intelligence (IJCAI)	
Conference on Neural Information Processing Systems (NeurIPS)	
Keywords: few-shot/one-shot learning, few-shot/one-shot classification, few-shot/one-shot image recognition	

Table 1: Selected conferences and journals (listed in alphabetical order of their abbreviations). Papers that include at least one of the keywords were considered for further investigation.

methods evolve in order to generalize better and be more applicable in the settings that mimic the reality more closely. Moreover, the rapid development of FSL leads to a considerable amount of methods proposed since the publications of [15] and [16]. These new approaches have been discussed in this review. [17] is the latest review on FSL published in 2021, but it is entirely devoted to meta-learning approaches and has very little overlap with our work. In short, this paper provides an up-to-date review of deep metric learning methods for few-shot image classification and a careful examination of different components of these methods to understand their strengths and limitations. The conferences and journals being surveyed are listed in Table 1. Papers that include at least one of the keywords are considered for further investigation on their relevance and contribution.

The rest of this review is organized as follows. Firstly for completeness, in Section 2 we give the definition of few-shot classification and introduce the evaluation procedure and commonly used datasets. Secondly, in Section 3 we review classical few-shot metric learning algorithms and recent influential works published from 2018 to 2022. In the light of the procedure of metric learning, these methods are classified into learning feature embeddings, learning class representations, and learning distance or similarity measures. Finally, we discuss some remaining challenges, future directions, and real-world applications in Section 4 and conclude this review in Section 5.

2. The framework of few-shot image classification

2.1. Notation and definitions

We first establish the notation and give a unified definition of various types of few-shot classification by generalizing the definition of few-shot learning [12].

Few-shot classification involves two datasets, **base dataset** and **novel dataset**. The novel dataset is the dataset on which the classification task is performed. The base dataset is an auxiliary dataset used to facilitate the learning of the classifier by transferring knowledge. We use $\mathbb{D}_{base} = \{(X_i, Y_i); X_i \in \mathcal{X}_{base}, Y_i \in \mathcal{Y}_{base}\}_{i=1}^{N_{base}}$ to denote the base dataset, where Y_i is the class label of instance X_i; in the case of image classification, X_i denotes the feature vector of the ith image. The novel dataset is denoted similarly by $\mathbb{D}_{novel} = \{(\tilde{X}_j, \tilde{Y}_j); \tilde{X}_j \in \mathcal{X}_{novel}, \tilde{Y}_j \in \mathcal{Y}_{novel}\}_{j=1}^{N_{novel}}$. $\mathbb{D}_{base}$ **and** $\mathbb{D}_{novel}$ **have no overlap in the label space**, i.e., $\mathcal{Y}_{base} \cap \mathcal{Y}_{novel} = \emptyset$. To train and test the classifier, we split $\mathbb{D}_{novel}$ into the support set $\mathbb{D}_S$ and the query set $\mathbb{D}_Q$.

Definition 1. Suppose the support set $\mathbb{D}_S$ is available, and the sample size of each class in $\mathbb{D}_S$ is very small (e.g., from 1 to 5). The **few-shot classification** task aims to learn from $\mathbb{D}_S$ a classifier $f : \mathcal{X}_{novel} \to \mathcal{Y}_{novel}$ that can correctly classify instances in the query set $\mathbb{D}_Q$. In particular, if $\mathbb{D}_S$ contains C classes and K labeled examples per class, the task is called **C-way K-shot classification**; if the sample size of each class in $\mathbb{D}_S$ is one, then the task is called **one-shot classification**.

Before presenting the next definition, we introduce the concept of domain. A *domain* consists of two components, namely a feature space $\mathcal{X}$ and a marginal distribution $P(X)$ over $\mathcal{X}$ [18].

Definition 2. A few-shot classification task is called **cross-domain few-shot classification** if the base dataset and the novel dataset come from two different domains, i.e., $\mathcal{X}_{base} \neq \mathcal{X}_{novel}$ or $P(X) \neq P(\tilde{X})$, where $X \in \mathcal{X}_{base}$ and $\tilde{X} \in \mathcal{X}_{novel}$.

Definition 3. The **generalized few-shot classification** task aims to learn a classifier $f : \mathcal{X}_{novel} \cup \mathcal{X}_{base} \to \mathcal{Y}_{novel} \cup \mathcal{Y}_{base}$ that can correctly classify instances in the query set $\mathbb{D}_Q$, where $\mathbb{D}_Q$ includes instance-label pairs from $\mathbb{D}_{base}$ in addition to existing pairs from $\mathbb{D}_{novel}$.

2.2. Evaluation procedure of few-shot classification

We provide a general procedure to evaluate the performance of a classifier for C-way K-shot classification in Algorithm 1. The evaluation procedure includes many episodes (i.e., tasks). In each episode, we first randomly select C classes from the novel label set, and then randomly select K samples from each of the C classes to form a support set and M samples from the remaining samples of those C classes to form a query set. Let $\mathbb{X}^{(e)}$ and $\mathbb{Y}^{(e)}$ denote the set of instances and the set of labels in the query set at the eth episode, respectively. A learning algorithm returns a classifier $f(\cdot|\mathbb{D}_{base}, \mathbb{D}_S^{(e)})$ upon

Algorithm 1 Evaluation procedure of C-way K-shot classification

Input: $\mathbb{D}_{base} = \{(X_i, Y_i); X_i \in \mathcal{X}_{base}, Y_i \in \mathcal{Y}_{base}\}_{i=1}^{N_{base}}$;$\mathbb{D}_{novel} = \{(\tilde{X}_j, \tilde{Y}_j);$ $\tilde{X}_j \in \mathcal{X}_{novel}, \tilde{Y}_j \in \mathcal{Y}_{novel}\}_{j=1}^{N_{novel}}$; number of episodes E.

for $e = 1, \cdots, E$ **do**

  Randomly select C classes from $\mathcal{Y}_{novel}$.

  Randomly select K samples from each class as the support set $\mathbb{D}_S^{(e)}$.

  Randomly select M samples from the remaining samples of C classes as the query set $\{(\mathbb{X}^{(e)}, \mathbb{Y}^{(e)})\}$.

  Record predicted labels $\hat{\mathbb{Y}}^{(e)} = f(\mathbb{X}^{(e)} | \mathbb{D}_{base}, \mathbb{D}_S^{(e)})$.

  Compute accuracy $a^{(e)} = \frac{1}{M}\sum_{j=1}^{M} \mathbb{1}[\hat{\mathbb{Y}}^{(e)} = \mathbb{Y}^{(e)}]$[a].

end for

return mean accuracy $\frac{1}{E}\sum_{e=1}^{E} a^{(e)}$.

[a] $\mathbb{1}$ denotes the element-wise indicator function.

receiving the base dataset and the eth support set, which predicts labels of query instances as $\hat{\mathbb{Y}}^{(e)} = f(\mathbb{X}^{(e)} | \mathbb{D}_{base}, \mathbb{D}_S^{(e)})$. Let $a^{(e)}$ denote the classification accuracy on the eth episode. The performance of a learning algorithm is measured by the classification accuracy averaged over all episodes.

2.3. Datasets for few-shot image classification

In this section, we briefly introduce benchmark datasets for few-shot image classification. Statistics of the datasets and commonly used experimental settings are listed below, and sample images are shown in Figure 1.

Omniglot [19]: one of the most widely used datasets for evaluating few-shot classification algorithms. It contains 1623 characters from 50 languages. The dataset is often augmented by rotations of 90, 180, 270 degrees, resulting in

6492 classes, which are split into 4112 base, 688 validation, and 1692 novel classes. The validation classes are used for model selection. The dataset is used less often in the latest studies as many methods can attain over 99% accuracy on the 5-way 1-shot classification task.

Mini-ImageNet and Tiered-ImageNet: another two widely used datasets derived from the ImageNet dataset [20]. Mini-ImageNet consists of 100 selected classes with 600 images for each class. This dataset was first proposed by Vinyals et al. [7], but recent studies follow the experimental setting provided by Ravi and Larochelle [21], which splits 100 classes into 64 base, 16 validation, and 20 novel classes. Tiered-ImageNet is a larger dataset with a hierarchical structure [22]. It is constructed from 34 super-classes with 608 classes in total and include 779,165 images. These super-classes are split into 20 base, 6 validation, and 8 novel super-classes, which correspond to 351 base, 97 validation, and 160 novel classes, respectively.

CIFAR-FS and FC100: two datasets derived from CIFAR-100 [23]. CIFAR-FS [24] contains 100 classes with 600 images per class, and it is split into 64 base, 16 validation, and 20 novel classes. FC100 [25] divides 100 classes into 20 super-classes, with five classes in each super-class. The dataset is split into 12 base, 4 validation, and 4 novel super-classes.

Stanford Dogs [26]: one of the benchmark datasets for fine-grained classification task, which contains 120 breeds (classes) of dogs with a total number of 20,580 images. These classes are divided into 70 base, 20 validation, and 30 novel classes.

CUB-200-2010/2011: another fine-grained dataset of 200 bird species. The

Figure 1: Sample images of some benchmark datasets for few-shot image classification. Datasets include Onimiglot, Mini-ImageNet, Fewshot-CIFAR100, Stanford Dogs, and CUB-200-2011.

initial version in 2010 collects 6033 images [27] and is extended in 2011 to 11,788 images [28]. The CUB-200-2010 dataset is commonly split into 130 base, 20 validation, and 50 novel classes [29], while the CUB-200-2011 dataset is commonly split into 100 base, 50 validation, and 50 novel classes [30].

Mini-ImageNet $\rightarrow$ *CUB*: a dataset used for cross-domain few-shot classification. Mini-ImageNet serves as the base dataset, 50 classes of CUB-200-2011 serve as the validation classes, and the remaining 50 classes serve as novel classes.

Meta-Dataset: a new, large-scale dataset for evaluating few-shot classification methods, particularly cross-domain methods. It initially consists of 10 diverse image datasets [31], e.g., ImageNet, CUB, and MS COCO [32], and later expanded with three additional datasets [33]. There are two training procedures and two evaluation protocols. In the more commonly used setting of training on all datasets (multi-domain learning) [33, 34, 35], the methods are trained on the official training splits of the first eight datasets, and they are evaluated on the test splits of the same datasets for in-domain performance and the remaining five datasets for out-of-domain performance. The other setting is training only on the Meta-Dataset version of ImageNet (single-domain learning), and evaluating on the test split of ImageNet for in-domain performance and the rest 12 datasets for out-of-domain performance.

3. Few-shot deep metric learning methods

The goal of supervised metric learning is to learn a distance metric to measure the similarity among samples such that it is optimal for the subsequent learning tasks. For example, for classification, samples from the same (different, resp.) class should be assigned with a small (large, resp.) distance. In the case of few-shot classification, the metric is learned on the base dataset; query images of the novel class are classified by computing their distances to novel support images with respect to the learned measure, followed by applying a distance-based classifier such as the k-nearest neighbor (kNN) algorithm. Traditional metric learning methods learn a Mahalanobis distance, which is equivalent to learning a linear transformation of original features [36]. However, in deep metric learning, the distance measure and feature embeddings are often learned separately so as to capture the nonlinear data structure and generate more discriminative feature representations. Moreover, instead of comparing with individual samples, many few-shot metric learning methods compare query samples with class representations such as prototypes and subspaces. In the remainder of this section, we provide a review of representative approaches, which are categorized into three groups according to the aspect they are improving on, namely 1) learning feature embeddings, 2) learning class representations, and 3) learning distance or similarity measures. A summary of these methods is provided in Figure 2.

3.1. Learning feature embeddings

Methods of learning feature embeddings implicitly assume that the network is powerful to extract discriminative features and can generalize well to novel classes. Early approaches aim at a task-agnostic embedding model that is effective for any task. More recently, endeavors are made to learn a task-specific embedding model for better distinguishing the classes at hand. Furthermore, techniques for data augmentation and multi-task learning are leveraged to address the issues of data scarcity and overfitting.

3.1.1. Learning task-agnostic features

The Siamese Convolutional Neural Network [6] is the first deep metric learning method for one-shot image classification. The Siamese Network, first introduced in [37], consists of two sub-networks with identical architectures and shared weights. [6] adopted the VGG-styled convolutional layers as the sub-network to extract high-level features from two images and employed the weighted L_1 distance as the distance between the two feature vectors. Weights of the network, as well as those of component-wise distance, are trained using the conventional technique of mini-batch gradient descent.

The Matching Network [7] encoded support and query images using

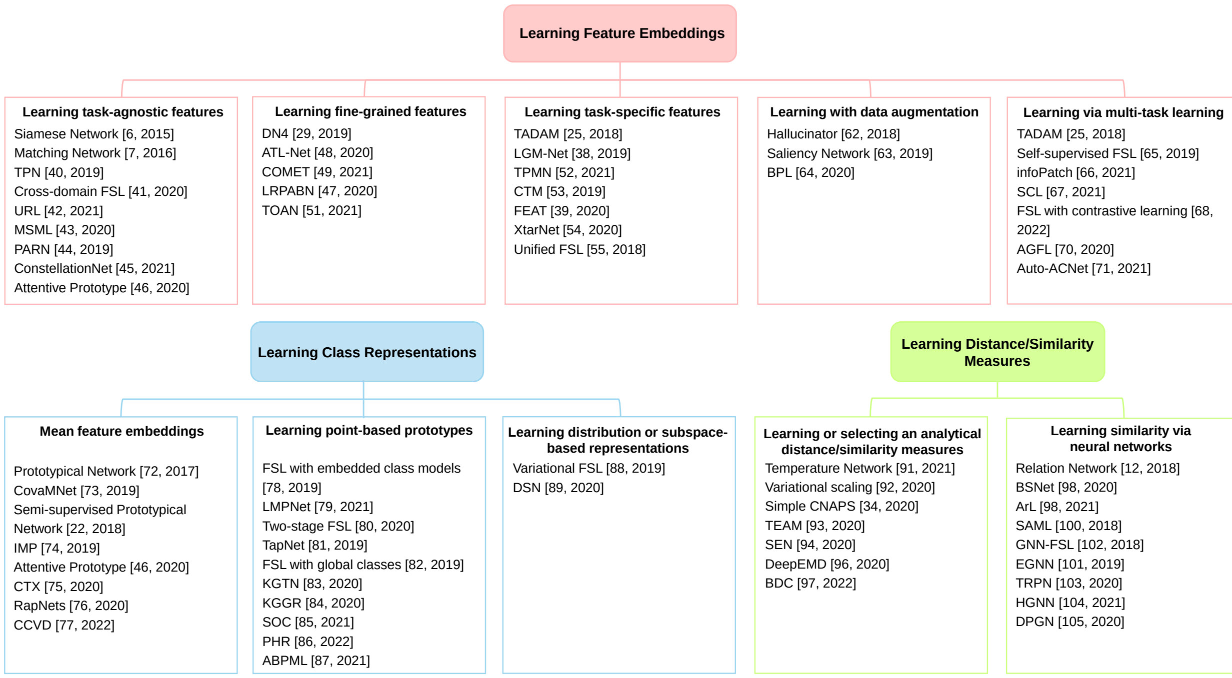

Figure 2: Taxonomy of few-shot deep metric learning methods reviewed in this paper. Some methods contribute to two aspects of metric learning and thus appear twice.

different networks in the context of the entire support set, and it first introduced episodic training to few-shot classification. A support image is embedded via a bidirectional LSTM network, which takes account of not only the image itself but also other images in the set; a query image is embedded via an LSTM with an attention mechanism to enable dependency on the support set. However, the sequential nature of bidirectional LSTM results in feature embeddings that will change with different ordering of samples in the support set. This issue can be sidestepped, such as by applying a pooling operation [38] or using self-attention [39]. The classification mechanism of Matching Network is suitable for few-shot learning. The network outputs a label distribution by computing a convex combination of one-hot label vectors of all support samples, with coefficients defined by using a softmax over cosine similarities; the class with the highest probability is selected as the predicted class. Another valuable contribution of [7] is the episode-based training strategy, which has been adopted by many subsequent works. Following meta-learning, the training phase on the base dataset should mimic the prediction phase where only few support samples are available. That is, gradient updates should be performed on episodes with C classes randomly sampled from the base label set and K examples for each class.

The episodic training strategy closes the gap between training and test distributions and thus alleviates the issue of overfitting to few labeled training images. The overfitting issue can be further addressed by utilizing query instances (i.e., excluding query labels) via transductive inference. Transductive Propagation Network (TPN) [40] is the first work adopting transductive inference for few-shot learning and introduced the idea of label propagation. Concretely, the network contains a feature embedding module and a graph construction module. The graph construction module, taking feature embeddings as inputs, learns a label propagation graph to exploit the manifold structure of support and query samples. Based on the learned kNN graph, labels are propagated from the support set to the query set; a closed-form solution of label propagation is used to speed up the prediction procedure. While transductive learning takes advantage of query instances, it is unsuitable for online learning where data arrive sequentially.

The aforementioned methods, designed for classifying novel data from the same domain, degrade when novel data comes from different domains [30]. Tseng et al. [41] noticed that this is caused by the large discrepancy between the feature distributions in different domains and proposed to simulate various feature distributions in the training stage as a general solution to enhance the domain generalization ability of metric learning methods. This is achieved by inserting multiple feature-wise transformation layers into the feature extractor; each transformation simulates one distribution, and the hyperparameters of affine transformations can be tuned via a meta-learning approach so that they are optimal to a particular metric learning method and capture the complex variation in feature distributions. Li et al. [42] proposed to learn a universal feature representation that works well for multiple domains. The technique of knowledge distillation is applied, where a multi-domain network is learned to generate universal features which align with features from multiple single-domain networks up to a linear transformation.

Motivated by the observation that the interested object may locate only in a region of an image and at different positions across images, a series of improvements on feature embedding have been proposed, such as by learning local features [29] and multi-scale features [43] and encoding the position information [44]. Local feature-based methods, while can be applied to generic few-shot image classification, are particularly effective for fine-grained image classification and thus will be discussed separately in the next subsection. Jiang et al. [43] proposed the Multi-Scale Metric Learning (MSML) network, which constructs multiple feature embeddings corresponding to different scales of the image. The similarity between support and query features at each scale is computed using the Relation Network (which will be introduced in Section 3.3.2). Wu et al. [44] proposed the Position-Aware Relation Network (PARN) to reduce the sensitivity of Relation Network to the spatial position of semantic objects. PARN adopts deformable convolutional layers to extract more effective features which filter out unrelated information like the background, and a dual correlation attention module to incorporate each spatial position of an image with the global information about the compared image and the image itself, so that the subsequent convolution operations, even subject to local connectivity, can perceive and compare semantic features in different positions. Compared with standard ways of overcoming position sensitivity, such as by using larger kernels or more layers, PARN is more parameter efficient. Xu et al. [45] proposed the ConstellationNet which extracts part-based features and encodes the spatial relationship between these representations by using self-attention with an explicit, learnable positional encoding. The spatial relationship between different parts of the image has also been encoded in [46] by using a capsule network.

3.1.2. Learning task-agnostic features for fine-grained image classification

Fine-grained image classification aims to distinguish different sub-categories under the same basic-level category. It is particularly challenging due to the subtle differences between different sub-categories and large variance in the same sub-category which may result from variations in the object's pose, scale, rotation, etc. Therefore, for effective classification, several methods have been proposed to extract local features and second-order features.

In deep nearest neighbor neural network (DN4) [29], the feature embedding module extracts multiple local descriptors from an image, which are essentially the feature maps learned via CNNs prior to adding the final image-level pooling layer. The classification is performed at an image-to-class level, meaning that the local descriptors from support images of the same class are put into one pool, kNNs in each class pool are searched for each query local descriptor, and the total distance over all local descriptors and kNNs is the distance between the query image and the corresponding class. The method is shown to be particularly effective on fine-grained datasets, and the idea of learning local descriptors has been adopted in other fine-grained classification methods [47]. The Adaptive Task-aware Local representations Network (ATL-Net) [48] improved DN4 by selecting local descriptors with learned thresholds and assigning them different weights based on episodic attention, which brings more flexibility than using kNNs and adjusts for the discriminability between classes, respectively. In contrast to learning one feature embedding over spatially local patches, COMET [49] learns multiple embedding functions over various parts of input features. A set of fixed binary masks, termed concepts, are applied to input features to separate an image into human-interpretable segments. For each concept, a feature embedding is learned to map masked features into a new discriminative feature space. The query image is classified according to the distances aggregated from all concept-specific spaces.

Huang et al. [47] proposed the Low-Rank Pairwise Alignment Bilinear Network (LRPABN) which aligns features spatially and extracts discriminative, second-order features. After learning first-order features from base images, the method trains a two-layer multi-layer perceptron network with two designed feature alignment losses to transform the positions of image features of a query image to match those of a support image, and designs a low-rank pairwise bilinear pooling layer which adapts the self-bilinear pooling [50] to extract second-order features from a pair of support and query images. The classification is performed as in the Relation Network. In the follow-up work, [51] improves the spatial alignment part by using the cross-channel attention to generate spatially matched support and query features and groups features in the convolutional channel dimension before the pooling layer as each group corresponds to a semantic concept.

3.1.3. Learning task-specific features

Methods reviewed in the preceding sections generate the same feature embedding for an image, regardless of the subsequent classification task. While this avoids the risk of overfitting, these generic features may not be sufficiently discriminative to distinguish novel classes. To this end, task-specific embedding models have been proposed to adapt features to the particular task; it should be noted that the adaptation is learned on the base dataset and does not involve any re-training on the novel dataset.

TADAM [25] is the first metric learning method which explicitly performs task adaptation. Exploiting the technique of conditional batch normalization, it applies a task-specific affine transformation to each convolutional layer of a task-agnostic feature extractor. The task is represented by the mean of class prototypes, and the scale and shift parameters of the affine transformation are generated from a separate network, called the Task Embedding Network (TEN). As TEN introduces more parameters and causes difficulty in optimization, the training scheme is revised to add the standard training, i.e., to distinguish all classes in the base dataset, as an auxiliary task to the episodic training.

Li et al. [38] proposed a meta-learning approach that can adapt weights of Matching Network to novel data. The proposed LGM-Net consists of a meta-learner termed MetaNet and a task-specific learner termed TargetNet.

The MetaNet module learns to produce a representation of each task from the support set and construct a mapping from the representation to weights of TargetNet. The TargetNet module, set as the Matching Network, embeds support and query images and performs classification. The proposed meta-learning strategy can be potentially implemented to adapt network parameters of other metric learning methods. Wu et al. [52] also proposed to learn task-specific parameters, but they combined the idea with local features. The proposed Task-aware Part Mining Network (TPMN) learns to generate parameters of filters used for extracting part-based features.

Different from the above two works which generate parameters for task-specific embedding layers, Li et al. [53] proposed to modify the generic features output from the task-agnostic embedding layers. A task-specific feature mask is generated from the Category Traversal Module (CTM), which includes a concentrator unit and a projector unit to extract features for intra-class commonality and inter-class uniqueness, respectively. It is noted that CTM can be easily embedded into most few-shot metric learning methods, such as Matching Network, Prototypical Network, and Relation Network; the latter two methods will be introduced in the following sections. Ye et al. [39] also proposed to adjust features directly, but instead of applying a mask, set-to-set functions are used to transform a set of task-agnostic features into a set of task-specific ones. These functions can model interactions between images in a set and hence enable co-adaptation of each image. Four set-to-set function approximators are presented in [39], and the one with Transformer, termed FEAT, is shown to be most effective.

Yoon et al. [54] proposed XtarNet to learn task-specific features for a new setting of generalized few-shot learning, where the model is trained on the base dataset, adapted given the support set of the novel dataset, and used to classify instances from both base and novel classes. XtarNet contains three meta-learners. The MetaCNN module adapts feature embeddings for each task. The MergeNet module produces weights for mixing pre-trained features and meta-learned features. As the classification is performed by comparing the mixed features with class prototypes, the TconNet module adapts prototypes of base and novel classes to improve discriminability. Rahman et al. [55] proposed a unified approach for zero-shot learning, generalized zero-shot learning and few-shot learning, which classifies a query image based on the similarity between its semantic representation and the textual features of each class. The semantic representation is a combination of two parts – one is a linear combination of base samples' semantic features, and the other one is based on the linear mapping learned from support images.

3.1.4. Feature learning with data augmentation

Data augmentation is a strategy that expands the support set in an artificial or model-based way with label preserving transformations, and thus is well-suited when the support samples are limited. One commonly used method is deformation [56, 57, 58], such as cropping, padding, and horizontal flipping. Besides this, generating more training samples [59, 60] and pseudo labels [61] are also popular techniques to augment data.

In few-shot learning, there is one class of works which places the data augmentation process into a model, that is, they embed a generator that can generate the augmented data to learn or imagine the diversity of data. Wang et al. [62] constructed an end-to-end few-shot learning method, in which the training data goes through two streams to output – one is from the original data to the classifier directly, and the other one is from the original data to a 'hallucination' network to augment data and then from the augmented data to classifier. Zhang et al. [63] developed a saliency-based data generation strategy. The Saliency Network obtains foregrounds and backgrounds of an image, which are used to achieve the hallucination for the image. In [64], a much simpler feature synthesis strategy was proposed, which synthesizes novel features by perturbing the semantic representations (i.e., word vectors of class labels) and projecting them into the visual feature space. In addition, when learning the projection function, a competitive learning formulation is adopted to push the synthesized sample towards the center of the most likely unseen class and away from that of the second best class.

3.1.5. Multi-task feature learning

Besides generating more training data, some works tried to exploit auxiliary information of samples to perform multi-task learning, which creates a regularization effect and helps learn discriminative features.

As briefly discussed above, TADAM [25] used an auxiliary task of training a normal global classifier on the base dataset to co-train the few-shot classifier; the task is sampled with a probability during the training process. An alternative auxiliary task is to exploit generative [65] or contrastive [66] self-supervised learning, which adopts self-defined pseudo labels as supervision to learn generalizable feature embeddings. In [65], support samples are artificially rotated to different number of degrees. A shared feature embedding is learned through two branches of networks, one for the original classification task and the other for identifying the rotation degree. In [66], infoPatch was proposed, which trains the embedding network episodically according to the standard classification loss and an auxiliary contrastive loss. The contrastive pairs are constructed for each query image, with the positive pair using support images of the same class and the negative pair using supports of different classes. To generate hard pairs, random blocks are applied to support images to mask parts of the image, and a query image is split into patches with one of them exchanged with a patch of another image. Not only in episodic training, contrastive learning can also be introduced in pre-training [67] or in both training stages [68]. In particular, in the episodic training stage of [68], the entire episode is regarded as the shared context, and two data augmentation strategies are applied to construct contrastive episodes. However, as noted by Xiao et al. [69], these contrastive learning methods require hand selecting augmentations and carefully tuning the hyperparameters to control the strength of augmentation. More severely, they implicitly assume invariance to particular transformations, e.g., rotation and color, which may be beneficial to some downstream tasks but harmful to others. One solution proposed in [69] is to use a multi-head network with a shared backbone to learn several embedding spaces, one for invariance to all augmentations and the others for invariance to all but one augmentation. The downstream task can flexibly utilize the optimal set of invariant features. The solution was proposed in a transfer learning setting; more research is needed for metric learning.

Zhu et al. [70] suggested that base and novel classes, despite being disjoint, can be connected by some visual attributes. Based on this insight, they used attribute learning as an auxiliary task. Visual attributes are provided as additional information during training, and the embedding network is learned to correctly predict both attribute labels and class labels. [71] also utilized attribute information, but in a richer way which requires an additional prediction of common and different attributes between an image pair. Moreover, the neural architecture search was first introduced to few-shot learning for automatically identifying the optimal operation from max pooling, convolution, identity mapping, etc for layers in the feature embedding network and attribute learning network.

3.2. Learning class representations

Early few-shot metric learning methods such as Siamese Network and Matching Network classify a query sample by measuring and comparing its distance to support samples. However, since support samples are scarce, they have limited capacity in representing the novel class. To alleviate this issue, some researchers propose to use class prototypes, which serve as reference vectors for each class. Prototypes can be constructed by taking simple or weighted average of feature embeddings, or learned in an end-to-end manner so as to further improve their representation ability. Besides point-based prototypes, some works consider the distribution of each class or use subspaces as class representations.

3.2.1. Feature embeddings-based prototypes

Prototypical Network [72] is a classical method that performs classification by calculating the Euclidean distance to class prototypes in the learned embedding space. It builds on the hypothesis that there exists an embedding space in which each class can be represented by a single prototype and all instances cluster around the prototype of their corresponding classes. In [72], the prototype of each class is set as the mean of feature embeddings of support samples in the class. Feature embeddings, and thus class prototypes, are learned using episodic training with the objective of minimizing the cross entropy loss. In [73], the class prototype is represented using the covariance matrix of feature embeddings. A covariance-based metric is also proposed to measure the similarity between the query and the class.

To make use of both labeled support samples and unlabeled samples, Ren et al. [22] proposed semi-supervised Prototypical Network, which is the first work of semi-supervised few-shot learning. The method adopts soft k-means to compute assignment score of unlabeled samples and computes the prototype as the mean of weighted samples based on assignment scores.

Considering that the dataset may exhibit multi-modality and multiple prototypes would be more suitable in this scenario, Infinite Mixture Prototypes (IMP) [74] was proposed to model multiple clusters in each class, and each cluster is modeled as a Gaussian distribution. Concretely, the probability that a sample follows the Gaussian distribution of each cluster determines which cluster the sample is assigned into. Moreover, the cluster variance of the Gaussian distributions, which needs to be learned, can affect the number of class prototype and performance of IMP.

Wu et al. [46] proposed to compute query-dependent prototypes. An attentive prototype is computed for each query as the weighted average of support samples and the weights are given by the Gaussian kernel with the Euclidean distance between the query and the support samples. As support samples that are more relevant to the query have greater influence on the classification, the method is more robust to outliers in support samples. Query-dependent prototypes have also been studied in CrossTransformers (CTX) [75], but they are computed separately for each spatial location. In other words, a local region of a query image is compared with an attentive prototype specific to this query and region, and the overall distance between the query and the prototype is the averaged distances over all local regions. Moreover, self-supervised episodes are constructed to train CTX.

Lu et al. [76] proposed the Robust attentive profile Networks (RapNets) to enhance the robustness of prototypes against outliers and label noises. The network transforms raw feature embeddings into correlation features in a nonparametric way and then inputs these features into a parametric bidirectional LSTM and fully-connected network to generate attention scores which serve as weights to combine support images. Moreover, training episodes are revised to include noisy data, and a new evaluation metric is proposed to evaluate the robustness of few-shot classification methods.

Ma et al. [77] provided a geometric interpretation of Prototypical Network, regarding it as a Voronoi diagram. In addition, the authors extended this perspective and proposed the Cluster-to-Cluster Voronoi Diagram (CCVD), which can ensemble models learned with different data augmentation, built on single or multiple feature transformations, and using linear or nearest neighbor classifier.

3.2.2. Point-based learnable prototypes

Ravichaandran et al. [78] adopted an implicit way to learn class representation instead of determining class prototypes as in the aforementioned methods. The prototype is modeled as a learnable and parameterized function of feature embedding of labeled samples in the class and is obtained by minimizing a loss which measures the distance between the feature embedding of a sample and the class prototype. Meanwhile, the function is shot free, that is, it allows sample sizes of classes in novel data to be unbalanced. In [79], prototypes are represented as weighted averages of feature embeddings, but different from [22, 46] discussed in the previous section, weights are learned end-to-end via episodic training. Moreover, instead of using image-level features, [79] combines local descriptors of one class following the idea of DN4 and learns multiple weight vectors to generate multiple prototypes per class. Das and Lee proposed a two-stage approach for generating class prototypes [80]. In the first stage, feature embeddings are learned, from which coarse prototypes of base and novel classes can be obtained from mean embeddings. In the second stage, the novel class prototype is refined through a meta-learnable function of its own prototype and related base prototypes.

Besides the above methods, TapNet [81] explicitly modeled class prototypes as learnable parameters. Prototypes and feature embeddings are learned simultaneously on the base dataset following the training procedure of Prototypical Network. In addition, to make prototypes and feature embeddings more specific to the current task, both of them are projected into a new classification space via a linear projection matrix. The projection matrix is obtained by using a linear nulling operation and does not include any learnable parameter. Luo et al. [82] proposed to learn prototypes of base and novel classes simultaneously by including the support set of novel classes in the training process. In each episode, local prototypes are generated from the sample synthesis module, which aims to increase the diversity of novel classes. They are then used in the registration module to update the global prototypes towards better separability. The query image is classified by searching for the nearest neighbor among global prototypes. As both base and novel prototypes are learned, the method can be readily applied to the generalized few-shot learning setting. Chen et al. [83] shared the same aim of learning base and novel prototypes, but additionally took advantage of the semantic correlations among these classes. A Knowledge Graph Transfer Network (KGTN) is proposed, which employs a gated graph neural network to represent class prototypes and correlations as nodes and edges, respectively. By propagating through the graph, information from correlated base classes is used to guide the learning of novel prototypes. This work is extended in [84] to the multi-label classification setting, which employs the attention mechanism and an additional graph for learning class-specific feature vectors. In [85], the Shared Object Concentrator (SOC) algorithm was proposed to learn a series of prototypes for each novel class from local features of support images. The first prototype is learned to have the largest cosine similarity with one of the local features, the second prototype has the second largest value, and so forth. The query image is classified according to the weighted sum of similarities between its local features and all prototypes, with weights decaying exponentially to account for the decreasing influence of prototypes. Zhou et al. [86] proposed the Progressive Hierarchical Refinement (PHR) method to update prototypes iteratively using all novel data. In each iteration, support images and a random subset of query images are embedded into features at local, global and semantic levels, and a loss function defined over these hierarchical features is used to refine prototypes for better inter-class separability. As each update is based on a random subset of queries, the method is less likely to overfit to noisy query samples, though it implicitly assumes the availability of a large number of queries.

Sun et al. [87] proposed to treat prototypes as random variables. The posterior distributions of latent class prototypes are learned by using amortized variational inference, a technique which enables prototype learning to be formulated as a probabilistic generative model without encountering severe computational and inferential difficulties.

3.2.3. Distribution or subspace-based representations

Considering that single point-based metric learning is sensitive to noise, Zhang et al. [88] proposed a variational Bayesian framework for few-shot learning and used the Kullback-Leibler divergence to measure the distance of samples. The framework can compute the confidence that a query image is assigned into each class by estimating the distribution of each class based on a neural network.

Simon et al. [89] proposed Deep Subspace Network (DSN) to represent each class using a low-dimensional subspace, constructed from support samples via singular value decomposition. Query samples are classified according to the nearest subspace classifier, that is to assign the query to the class which has the shortest Euclidean distance between the query and its projection onto the class-specific subspace. The method is shown to be more robust to noises and outliers than Prototypical Network.

3.3. Learning distance or similarity measures

Methods reviewed in Sections 3.1 and 3.2 focus on learning a discriminative feature embedding or obtaining an accurate class representation. For classification, they mostly adopt a fixed distance or similarity measure, such as the Euclidean distance [72] and the cosine similarity [7]. More recently, researchers propose to learn parameters in these fixed measures or define novel measures so as to further improve the classification accuracy. Moreover, considerable effort has been made to learn similarity scores by using fully-connected neural networks or Graph Neural Networks (GNNs).

3.3.1. Learning or selecting an analytical distance or similarity measure

In TADAM [25], Oreshkin et al. mathematically analyzed the effect of metric scaling on the loss function. Since then, many works tune the scaling parameter via cross-validation [48, 90]. Zhu et al. [91] proposed to use two different scaling parameters for the ground-truth class and other classes to enforce the same-class distance is much smaller than the different-classes distance. Moreover, the scaling parameters are gradually tuned every few episodes, which implements the idea of self-paced learning to learn from easy to hard. Chen et al. [92] proposed to learn the scaling parameter in a Bayesian framework. By assuming a univariate or multivariate Gaussian prior and applying the stochastic variational inference technique for approximating the posterior distribution, a scaling parameter or a scaling vector can be learned respectively, which scales the distance equally over all dimensions or differently for each dimension. Task-specific scaling vectors can also be learned by learning a neural network from the task to variational parameters.

The traditional Mahalanobis distance decorrelates and scales features using the inverse of the covariance matrix. In Simple CNAPS [34], after extracting features using the architecture of Conditional Neural Adaptive Processes

(CNAPS) [33], the classification is performed based on the Mahalanobis distance between query instances and class prototypes. Task-specific class-specific covariance matrices are estimated as convex combinations of sample covariance matrices estimated from instances of the task and instances of the class and regularized toward an identity matrix. Transductive Episodic-wise Adaptive Metric (TEAM) [93] learned task-specific metric from support and query samples. TEAM contains three modules, namely a feature extractor, a task-specific metric module, and a similarity computation module. The task-specific metric module learns a Mahalanobis distance to shrink the distance between similar pairs and enlarge the distance between dissimilar pairs, following the objective function of the pioneering metric learning method [36].

Nguyen et al. [94] proposed a dissimilarity measure termed SEN, which combines the Euclidean distance and the difference in the L_2-norm. Minimizing this measure will encourage feature normalization and consequently benefit the classification performance [95]. DeepEMD [96] combined a structural distance over dense image representations, Earth Mover's Distance (EMD) and convolutional feature embedding to conduct few-shot learning. The optimal matching flow parameters in EMD and the parameters in the feature embedding are trained in an end-to-end fashion. Xie et al. [97] introduced the Brownian Distance Covariance (BDC) metric, a new distance measure founded on the characteristic function of random vectors. The metric has a closed-form expression for discrete feature vectors and can be computed easily by first computing the BDC matrix for every image and then calculating the inner product between two BDC matrices. The computation of BDC matrices also only involves standard matrix operations and can be formulated as a pooling layer, thus endowing the method with high computational efficiency and ease of integrating with other few-shot classification methods.

3.3.2. Learning similarity scores via neural networks

The Relation Network [12] is the first work of introducing a neural network to model the similarity of feature embeddings in few-shot learning. It consists of an embedding module and a relation module. The embedding module builds on convolutional blocks for mapping original images into an embedding space, and the relation module consists of two convolutional blocks and two fully-connected layers for computing the similarity between each pair of support and query images. The learnable similarity measure enhances the model flexibility. Li et al. [98] pointed out that a single similarity measure may not be sufficient to learn discriminative features for fine-grained image classification and thus proposed the Bi-Similarity Network (BSNet), which integrates the proposed cosine module with existing similarity measures such as the relation module, forcing features to adapt to two similarity measures of diverse characteristics and consequently generating a more compact feature space. In principle, the method can be further developed to ensemble multiple metrics, and more importantly, an elegant way to determine the optimal set of metrics to be combined is needed. Relation Network and subsequent methods all use class labels to form binary supervision, indicating whether the image pair comes from the same class. Zhang et al. [99] argued that such binary relations are not sufficient to capture the similarity nuance in the real-world setting and therefore proposed a new method termed Absolute-relative Learning (ArL) which, in addition to binary relations, constructs continuous-valued relations from attributes of images, such as colors and textures.

Different from Relation Network, Semantic Alignment Metric Learning (SAML) [100] adopted the Multi-Layer Perceptron (MLP) network for computing the similarity score. Specifically, SAML contains a feature embedding module and a semantic alignment module. In the semantic alignment module, a relation matrix at the level of local features is first computed by using fixed similarity measures and an attention mechanism, and then fed into a MLP network which outputs the similarity score between the query and the support class. Due to the use of relation matrix as the input, the MLP network has more parameters than Relation Network, thus increasing the risk of overfitting.

Recently, some researchers adopt Graph Neural Networks (GNNs) to implement few-shot classification. Like the above reviewed works, GNN-based methods also use a neural network to model the similarity measure, while its advantage lies in the rich relational structure on samples [101]. Garcia et al. [102] proposed the first GNN-based neural network for few-shot learning, short for GNN-FSL here. It contains two modules, a feature embedding module and a GNN module. In the GNN module, a node represents a sample, and more specifically, equals the concatenation of features of the sample and its label. For a query sample, its initial label in the first GNN layer uses uniform distribution over K-simplex (K is number of classes), and its predicted label in the last GNN layer is used for computing the loss function. Like GNN-FSL, Edge-labeling Graph Neural Network (EGNN) [101] also contains a feature embedding module and a GNN module with three layers. However, rather than labeling nodes, EGNN learns to label edges in

GNN layers so that it can cluster samples explicitly by employing the intra-cluster similarity and inter-cluster dissimilarity. In EGNN, each GNN layer has its own loss function that is computed based on edge values in the layer, and the total loss is the weighted sum of loss functions of all GNN layers. The Transductive Relation-Propagation graph neural Network (TRPN) [103] explicitly modeled the relation of support-query pairs by treating them as graph nodes. After relation propagation, a similarity function is learned to map the updated node to a similarity score, which represents the probability that the support and query samples are of the same class. The class with the highest sum of scores is the predicted class. The Hierarchical Graph Neural Network (HGNN) [104], aimed at modeling the hierarchical structure within classes, first down-samples support nodes to build a hierarchy of graphs and then performs up-sampling to reconstruct all support nodes for prediction.

The previous GNN-based methods focus simply on the relation between a pair of samples. In Distribution Propagation Graph Network (DPGN) [105], the global relation between a sample and all support samples is considered by generating a distribution feature from the similarity vector. A dual complete graph is built to proceed sample-level and distribution-level features independently, and a cyclic update policy is used to propagate between the two graphs. Information from the distribution graph refines sample-level node features and hence improves the classification based on edge similarities.

Table 2 summarizes few-shot deep metric learning methods, listing the backbone network for feature embedding, classification mechanism, similarity measure, training strategy, datasets studied in the experiment, and classification performance. As the methods were implemented with different backbone networks and tested on different datasets, for a fair comparison, we select Conv-4 and ResNet-12 backbones whenever possible and report the 5-way 1-shot and 5-way 5-shot classification accuracy on Mini-ImageNet. Moreover, we notice that some methods were trained with higher ways or higher shots, which may lead to better performance, and thus this information is included under training strategy. Nevertheless, there are other factors which may affect the performance, such as the use of data augmentation techniques, optimization strategy, and the number of test episodes. Table 3 is a summary for few-shot fine-grained image classification. Here we note that the CUB dataset was split into training, validation and test sets in multiple ways.

4. Further research

Even though few-shot metric learning methods have achieved the promising performance, there remains several important challenges that need to be dealt with in the future. In this section, we will discuss issues related to generalization and robustness of few-shot learning methods, training strategy, and applicability, as well as listing some promising applications of few-shot metric learning methods.

4.1. Challenges and future directions

1. Improving generalized feature learning on few samples. In the existing few-shot metric learning methods or even the entire few-shot learning methods, researchers mostly try to learn discriminative feature based on the attention mechanism, data augmentation, multi-task learning, and so on. To learn feature with good generalization ability from few labeled examples, new ways of evaluation and feature learning need to be developed.

Table 2: Summary of deep metric learning methods for few-shot image classification.

Method	Classification mechanism	Similarity measure	Training strategies	Mini-ImageNet (Conv-4) 1-shot	Mini-ImageNet (Conv-4) 5-shot	Mini-ImageNet (ResNet-12) 1-shot	Mini-ImageNet (ResNet-12) 5-shot	Additional datasets or embedding architectures
Siamese Network [6]	w.r.t. instances	weighted L_1 distance	minibatch training	-	-	-	-	Omniglot
Matching Network [7]	w.r.t. instances	cosine similarity	episodic training	46.60	60.00	-	-	Omniglot
TPN [40]	w.r.t. instances	weighted Euclidean distance (learnable weights)	episodic training (higher-shot training)	55.51	69.86 (T)	-	-	Tiered-ImageNet
Cross-domain FSL [41]	w.r.t. instances	learned distance	pre-train + episodic training	-	-	66.32 ± 0.80 (ResNet-10)	81.98 ± 0.55 (ResNet-10)	-
URL [42]	w.r.t. prototypes	cosine similarity	episodic training	-	-	-	-	Meta-Dataset
MSML [43]	w.r.t. prototypes	learned distance	pre-train + episodic training	-	-	72.41 ± 1.72 (ResNet-50)	84.33 ± 1.14 (ResNet-50)	Tiered-ImageNet
PARN [44]	w.r.t. instances	learned distance	episodic training	55.22 ± 0.84	71.55 ± 0.66	-	-	Omniglot
ConstellationNet [45]	w.r.t. prototypes	cosine similarity	episodic training	58.82 ± 0.23	75.00 ± 0.18	64.89 ± 0.23	79.95 ± 0.17	CIFAR-FS, FC100
TADAM [25]	w.r.t. prototypes	Euclidean distance	episodic training + co-training	-	-	58.5 ± 0.3	76.7 ± 0.3	FC100
LGM-Net [38]	w.r.t. instances	cosine similarity	episodic training	69.13 ± 0.35	71.18 ± 0.68	-	-	Omniglot
TPMN [52]	w.r.t. prototypes	weighted sum of dot products	pre-train + episodic training	-	-	67.64 ± 0.63	83.44 ± 0.43	Tiered-ImageNet, CIFAR-FS, FC100
CTM [53]	w.r.t. instances or prototypes	Any, e.g., cosine similarity Euclidean, learned distance	pre-train (opt.) + episodic training	-	-	64.12 ± 0.82 (ResNet-18)	80.51 ± 0.13 (ResNet-18)	Tiered-ImageNet
FEAT [39]	w.r.t. instances	cosine similarity	pre-train + fine-tune temperature scaling	55.15 ± 0.20	71.61 ± 0.16	66.78 ± 0.20	82.05 ± 0.14	Tiered-ImageNet, OfficeHome; WRN
Hallucinator [62]	w.r.t. prototypes	cosine similarity	episodic training	-	-	-	-	ImageNet; ResNet-10, ResNet-50
Saliency Network [63]	w.r.t. instances	learned distance	episodic training	57.45 ± 0.88	72.01 ± 0.67	-	-	Open MIC
BPL [64]	w.r.t. prototypes	Euclidean distance with learned projection matrix	pre-train + episodic training	54.20 ± 0.58	65.28 ± 0.59	59.57 ± 0.63	76.86 ± 0.49	on ZSL, GZSL; WRN
Self-supervised FSL [65]	w.r.t. prototypes	Euclidean distance	episodic training / minibatch training	54.83 ± 0.43	71.86 ± 0.33	62.93 ± 0.46 (WRN)	79.87 ± 0.33 (WRN)	Tiered-ImageNet, ImageNet-FS; Conv-4-512, ResNet-10
infoPatch [66]	w.r.t. instances	cosine similarity	episodic training	-	-	67.67 ± 0.45	82.44 ± 0.31	Tiered-ImageNet, FC100
SCL [67]	w.r.t. prototypes	Euclidean distance	pre-train / episodic training	-	-	-	77.60 (ResNet-18)	Tiered-ImageNet; CIFAR-FS, FC100 ResNet-12 (for transfer learning)
FSL with contrastive learning [68]	w.r.t. prototypes	Euclidean distance	pre-train + episodic training	-	-	70.19 ± 0.46	84.66 ± 0.29	Tiered-ImageNet, CIFAR-FS
AGFL [70]	w.r.t. instances or prototypes	Any, e.g., cosine similarity, Euclidean, learned distance	episodic training	-	-	56.59 ± 0.64 (ResNet-50)	73.58 ± 0.48 (ResNet-50)	CUB-200-2011, AwA
Prototypical Network [72]	w.r.t. prototypes	Euclidean distance	episodic training (higher-way training)	49.42 ± 0.78	68.20 ± 0.66	-	-	Omniglot; CUB-200-2011 (for ZSL)
Semi-supervised Prototypical Network [22]	w.r.t. prototypes	Euclidean distance	episodic training	50.41 ± 0.31	64.39 ± 0.24	-	-	Omniglot, Tiered-ImageNet
IMP [74]	w.r.t. prototypes	Euclidean distance	episodic training	49.60 ± 0.80	68.10 ± 0.80	-	-	Omniglot, Tiered-ImageNet
Attentive Prototype [46]	w.r.t. prototypes	Euclidean distance	episodic training	-	-	66.43 ± 0.26 (DeepCaps)	82.13 ± 0.21 (DeepCaps)	Tiered-ImageNet, FC100
CTX [75]	w.r.t. prototypes	Euclidean distance	episodic training	-	-	-	-	Meta-Dataset; ResNet-34
RapNets [76]	w.r.t. prototypes	Euclidean distance	episodic training (higher-way training)	-	70.89 ± 0.64	-	-	Omniglot, FC100, CUB-200-2011

Table 2 (cont.)

Method	Classification mechanism	Similarity measure	Training strategies	Mini-ImageNet (Conv-4) 1-shot	Mini-ImageNet (Conv-4) 5-shot	Mini-ImageNet (ResNet-12) 1-shot	Mini-ImageNet (ResNet-12) 5-shot	Additional architectures or datasets
CCVD [77]	w.r.t. prototypes	Euclidean distance	episodic training	48.47 ± 0.86	65.86 ± 0.73	69.48 ± 0.45 (WRN)	86.75 ± 0.28 (WRN)	Tiered-ImageNet, CUB-200-2011; MobileNet, ResNet-10/18/34, DenseNet-121
FSL with embedded class models [78]	w.r.t. prototypes	Euclidean distance with learned projection matrix	episodic training	49.07 ± 0.43	65.73 ± 0.36	59.00	77.46	Tiered-ImageNet, CIFAR-FS
LMPNet [79]	w.r.t. prototypes	cosine similarity	episodic training	49.87 ± 0.20	68.81 ± 0.16	62.74 ± 0.11	80.23 ± 0.52	Tiered-ImageNet, CUB-200-2010, Stanford Dogs, Stanford Cars
Two-Stage FSL [80]	w.r.t. prototypes	Mahalanobis distance	episodic training (higher-way training in the first stage)	52.68 ± 0.51	70.91 ± 0.85	-	-	Omniglot, CIFAR-FS, CUB-200-2011
TapNet [81]	w.r.t. prototypes	Mahalanobis distance	episodic training (higher-way training)	50.68 ± 0.11	69.00 ± 0.09	61.65 ± 0.15	76.36 ± 0.10	Omniglot, Tiered-ImageNet
FSL with global class representations [82]	w.r.t. prototypes	Euclidean distance	pre-train + episodic training	53.21 ± 0.40	72.34 ± 0.32	-	-	Omniglot
KGTN [83]	w.r.t. prototypes	dot product, cosine similarity, Pearson correlation coefficient	pre-train + minibatch training	-	-	-	-	ImageNet-FS, ImageNet-6K; ResNet-50
SOC [85]	w.r.t. prototypes	cosine similarity	pre-train + fine-tune/episodic training of feature embeddings	-	-	69.28 ± 0.49	85.16 ± 0.42	Tiered-ImageNet
PHR [86]	w.r.t. prototypes	Euclidean distance	pre-train + episodic training	65.10 ± 0.70	78.10 ± 0.40	74.90 ± 0.60	84.50 ± 0.30	CIFAR-FS, FC100, CUB-200-2011; ResNet-18
ABPML [87]	w.r.t. prototypes	probability from Gaussian distribution	episodic training	53.28 ± 0.91	70.44 ± 0.72	-	-	Omniglot, CUB-200-2011, Stanford Dogs
Variational FSL [88]	w.r.t. distributions	probability from Gaussian distribution	pre-train + episodic training	57.15 ± 0.31	71.54 ± 0.23	61.23 ± 0.26	77.69 ± 0.17	Omniglot; cluttered Omniglot (for segmentation)
DSN [89]	w.r.t. subspaces	Euclidean distance	episodic training	55.88 ± 0.90	70.50 ± 0.68	64.60 ± 0.72	79.51 ± 0.50	Tiered-ImageNet, CIFAR-FS, Open MIC
Temperature Network [91]	w.r.t. prototypes	Euclidean distance	episode training	52.39	67.89	-	-	Stanford Dogs, Stanford Cars, Dermnet skin disease
Variational scaling [92]	w.r.t. prototypes	Euclidean distance or cosine similarity	episodic training	49.34 ± 0.29	67.83 ± 0.16	56.09 ± 0.19	74.46 ± 0.17	
Simple CNAPS [34]	w.r.t. prototypes	Mahalanobis distance	pre-train	-	-	82.16 evaluation over 5 runs only	89.80 (ResNet-18) evaluation over 5 runs only	Tiered-ImageNet, Meta-Dataset
TEAM [93]	w.r.t. prototypes	Mahalanobis distance	pre-train (for Conv-4 only) + episodic training	56.57	72.04 (T)	60.07 (ResNet-18)	75.90 (T) (ResNet-18)	CIFAR-FS, CUB-200-2011
SEN [94]	w.r.t. prototypes	SEN dissimilarity measure	episodic training	-	69.80	-	72.3 (WRN-16-6)	Omniglot, FC100
DeepEMD [96]	w.r.t. prototypes	Earth Mover's Distance	pre-train + episodic training	-	-	65.91 ± 0.82	82.41 ± 0.56	Tiered-ImageNet, FC100, CUB-200-2011
DeepBDC [97]	w.r.t. prototypes	Brownian Distance Covariance metric	pre-train / episodic training	-	-	67.83 ± 0.43	85.45 ± 0.29	Tiered-ImageNet, CUB-200-2011
Relation Network [12]	w.r.t. prototypes	learned distance	episodic training	50.44 ± 0.82	65.32 ± 0.70	-	-	Omniglot; AwA, CUB-200-2011 (for ZSL)
ArL [99]	w.r.t. prototypes	learned distance	episodic training	57.48 ± 0.65	72.64 ± 0.45	65.21 ± 0.58	80.41 ± 0.49	CUB-200-2011, Flowers
SAML [100]	w.r.t. prototypes	learned distance	episodic training	57.69 ± 0.2	73.03 ± 0.16	-	-	CUB-200-2011

Table 2 (cont.)

Method	Classification mechanism	Similarity measure	Training strategies	Mini-ImageNet (Conv-4) 1-shot	Mini-ImageNet (Conv-4) 5-shot	Mini-ImageNet (ResNet-12) 1-shot	Mini-ImageNet (ResNet-12) 5-shot	Additional architectures or datasets
GNN-FSL [102]	w.r.t. instances	learned distance	episodic training	50.33 ± 0.36	66.41 ± 0.63	-	-	Omniglot
EGNN [101]	w.r.t. instances	learned distance	episodic training	-	76.37 (T)	-	-	Tiered-ImageNet
TRPN [103]	w.r.t. instances	learned distance	episodic training	57.84 ± 0.51	78.57 ± 0.44 (T)	68.25 ± 0.50 (WRN)	85.40 ± 0.39 (T) (WRN)	Tiered-ImageNet
HGNN [104]	w.r.t. instances	learned distance	episodic training	60.03 ± 0.51	79.64 ± 0.36 (T)			Tiered-ImageNet, CUB-200-2011
DPGN [105]	w.r.t. instances	learned distance	episodic training	66.01 ± 0.36	82.83 ± 0.41 (T)	67.77 ± 0.32	84.60 ± 0.43 (T)	Tiered-ImageNet, CIFAR-FS, CUB-200-2011; ResNet-18, WRN

All experimental results are reported for 5-way classification. (T) denotes transductive setting. Unless specified otherwise, Conv-4 uses 4 convolutional layer with 64 filters, and WRN uses 28 convolutional layers with a widening factor of 10.

Table 3: Summary of deep metric learning methods for few-shot fine-grained image classification.

Method	Classification mechanism	Similarity measure	Training strategies	CUB-200-2011 1-shot	CUB-200-2011 5-shot	Stanford Dogs 1-shot	Stanford Dogs 5-shot	Additional datasets or embedding architectures
DN4 [29]	w.r.t. bags of local features	cosine similarity	episodic training	53.15±0.84 (CUB-2010)	81.90±0.60 (CUB-2010)	45.73±0.76	66.33±0.66	Mini-ImageNet, Stanford Cars
ATL-Net [48]	w.r.t. bags of local features	cosine similarity & learned distance	episodic training	60.91±0.91 (CUB-2010)	77.05±0.67 (CUB-2010)	54.49±0.92	73.20±0.69	Mini-ImageNet, Stanford Cars
CovaMNet [73]	w.r.t. bags of local features	covariance metric	episodic training	52.42±0.76 (CUB-2010)	63.76±0.64 (CUB-2010)	49.10±0.76	63.04±0.65	Mini-ImageNet, Stanford Cars
COMET [49]	w.r.t. prototypes	Euclidean distance	episodic training	67.9 ± 0.9	85.3 ± 0.5	-	-	Flowers; Conv-6
LRPABN [47]	w.r.t. prototypes	learned distance	episodic training	63.63±0.77 (120/30/50)	76.06±0.58 (120/30/50)	45.72±0.75	60.94±0.66	Stanford Cars
TOAN [51]	w.r.t. prototypes	learned distance	episodic training	65.34 ± 0.75 (120/30/50)	80.43 ± 0.60 (120/30/50)	51.83 ± 0.80	69.83 ± 0.66	Stanford Cars; ResNet-12
Auto-ACNet [71]	w.r.t. instances	learned distance	minibatch training	57.93 ± 0.54 (80/40/80; ACNet)	70.42 ± 0.43 (80/40/80; ACNet)	-	-	AwA2
BSNet [98]	w.r.t. prototypes	cosine similarity + learned distance	episodic training	62.84±0.95	85.39±0.56	43.42±0.86	71.90±0.68	Stanford Cars

All experimental results are reported for 5-way classification with Conv-4-64 as the embedding architecture. Unless specified otherwise, CUB-200-2011 [28] is split into 100/50/50 training/validation/test sets. CUB-2010 refers to CUB-200-2010 [27] with the split of 120/30/50. For Stanford Dogs, the dataset is split into 70/20/30 training/validation/test sets.

2. Enhancing stability to support samples and robustness to adversarial perturbations and distribution shifts. Despite the continuous improvement in classification accuracy, few-shot classification methods are vulnerable in various scenarios, hindering their usage in safety-critical applications such as medical image analysis. Prior works show that existing methods are non-robust to input or label outliers [76], adversarial perturbations (i.e., small, visually imperceptible changes of data that fool the classifier to make incorrect predictions) added to support [106] or query images [107], and distribution shift between support and query datasets [108]. In [109], it is demonstrated that even non-perturbed and in-distribution support images can significantly deteriorate the classification accuracy of several popular methods. Further exploration of vulnerability in existing approaches and design of robust and stable models will be very valuable.

3. Rethinking the use of episodic training strategy. While episodic training is a common practice to train metric learning methods in the few-shot learning setting, it is rigid to require each training episode to have the same number of classes and images as the evaluation episode; in fact, [72] observed the benefit of training with a larger number of classes. Moreover, the model gets updated after receiving an episode without regard to its quality and thus is prone to poorly sampled images like outliers. [110] is the first attempt to alleviate this problem by exploiting the relationship between episodes; more solutions are needed to identify episodes that are high-quality and useful to the novel task. Furthermore, we notice that episodic training can result in models that underfit the base dataset. One possible reason is that, by using episodic training, methods adopt continual learning on plenty of tasks sampled from the base dataset and suffer from catastrophic forgetting [111, 112], i.e., the model learned from previous tasks is supplanted after learning on a new task. Therefore, how to avoid this problem and enhance the model fitting ability of metric learning methods on both base and novel datasets remains a challenge.

4. Developing metric learning methods for cross-domain few-shot classification. While base and novel datasets may come from different domains in practice, currently only few works focus on cross-domain few-shot classification. More recently and severely, [113] reported that all meta-trained methods, including the reviewed work [41], are outperformed by the simple transductive fine-tuning in the presence of a large domain shift, specifically, when training on natural images and evaluating beyond them, such as on agriculture and satellite images. The difficulty is that the base data and the novel data usually have different metric spaces. Therefore, how to alleviate domain shift between the training and evaluation phases needs to be explored in the future.

4.2. Applications

The superior performance of deep metric learning methods for few-shot image classification motivates researchers to extend these methods to non-natural images from various disciplines. For example, the methods have been developed for diagnosing and classifying diseases based on dermoscopic [114] images and computerised tomography (CT) images [115], classifying plant diseases based on leaf images collected in the field [116], scene classification in aerial images [117] and remote sensing images [118], and hyperspectral image classification [119, 120].

Deep metric learning has also been applied beyond image classification, to more challenging computer vision applications. A notable example is person re-identification (Re-ID), whose aim is to retrieve a person of interest across multiple non-overlapping cameras [121, 122]. Metric learning is particularly effective for Re-ID, as this is an open-set classification task with different people in the training and test classes and often there is only one image available for the query person [123]. Metric learning also shows impressive results on face recognition, in both closed-set [124] and open-set [125] settings, and content-based image retrieval [126, 127], which can be formulated as a ranking problem.

5. Conclusions

This paper presents a review of recent few-shot deep metric learning methods. After providing the definitions and a general evaluation framework for few-shot learning and expounding on the widely used datasets and their settings, we review the novelty and limitations of existing methods. In particular, there is a pattern of progressing towards learning task-specific feature embeddings, task-dependent prototypes, and more flexible similarity measures. In addition, we list applications where few-shot deep metric learning prevails and suggest future research on improving feature generalizability, method robustness, training strategy, and applicability to cross-domain settings.

Acknowledgements

This work was supported in part by the Beijing Natural Science Foundation under Grant Z200002, the Royal Society under International Exchanges Award IEC\NSFC\201071, the National Natural Science Foundation of China (NSFC) under Grant 62111530146, 62176110, 61906080, 61922015, U19B2036, 62225601, and Young Doctoral Fund of Education Department of Gansu Province under Grant 2021QB-038, and Hong-liu Distinguished Young Talents Foundation of Lanzhou University of Technology.

References

[1] A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.

[2] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014).

[3] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.

[4] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, Recent advances in convolutional neural networks, arXiv preprint arXiv:1512.07108 (2015).

[5] F.-F. Li, R. Fergus, P. Perona, One-shot learning of object categories, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (4) (2006) 594–611.

[6] G. Koch, R. Zemel, R. Salakhutdinov, Siamese neural networks for one-shot image recognition, in: International Conference on Machine Learning deep learning workshop, Vol. 2, 2015.

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

[8] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, T. Lillicrap, One-shot learning with memory-augmented neural networks. arxiv preprint, arXiv preprint arXiv:1605.06065 (2016).

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

[10] M. Rohrbach, S. Ebert, B. Schiele, Transfer learning in a transductive setting, in: Advances in Neural Information Processing Systems, 2013, pp. 46–54.

[11] Q. Sun, Y. Liu, T.-S. Chua, B. Schiele, Meta-transfer learning for few-shot learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 403–412.

[12] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, T. M. Hospedales, Learning to compare: Relation network for few-shot learning, in: IEEE

Conference on Computer Vision and Pattern Recognition, 2018, pp. 1199–1208.

[13] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, S. X. Yu, Large-scale long-tailed recognition in an open world, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2537–2546.

[14] J. Shu, Z. Xu, D. Meng, Small sample learning in big data era, arXiv preprint arXiv:1808.04572 (2018).

[15] Y. Wang, Q. Yao, J. T. Kwok, L. M. Ni, Generalizing from a few examples: A survey on few-shot learning, ACM Computing Surveys 53 (3) (2020) 1–34.

[16] J. Lu, P. Gong, J. Ye, C. Zhang, Learning from very few samples: A survey, arXiv preprint arXiv:2009.02653 (2020).

[17] X. Li, Z. Sun, J.-H. Xue, Z. Ma, A concise review of recent few-shot meta-learning methods, Neurocomputing 456 (2021) 463–468.

[18] S. J. Pan, Q. Yang, A survey on transfer learning, IEEE Transactions on Knowledge and Data Engineering 22 (10) (2009) 1345–1359.

[19] B. Lake, R. Salakhutdinov, J. Gross, J. Tenenbaum, One shot learning of simple visual concepts, in: Proceedings of the annual meeting of the cognitive science society, Vol. 33, 2011.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., ImageNet

large scale visual recognition challenge, International Journal of Computer Vision 115 (3) (2015) 211–252.

[21] S. Ravi, H. Larochelle, Optimization as a model for few-shot learning, International Conference on Learning Representations (2017).

[22] 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, International Conference on Learning Representations (2018).

[23] A. Krizhevsky, Learning multiple layers of features from tiny images, University of Toronto (2009).

[24] L. Bertinetto, J. F. Henriques, P. H. Torr, A. Vedaldi, Meta-learning with differentiable closed-form solvers, International Conference on Learning Representations (2019).

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

[26] A. Khosla, N. Jayadevaprakash, B. Yao, F.-F. Li, Novel dataset for fine-grained image categorization: Stanford dogs, in: CVPR Workshop on Fine-Grained Visual Categorization, Vol. 2, 2011.

[27] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, P. Perona, Caltech-UCSD birds 200 (2010).

[28] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, The Caltech-UCSD birds-200-2011 dataset (2011).

[29] 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: IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[30] 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.

[31] E. Triantafillou, T. Zhu, V. Dumoulin, P. Lamblin, U. Evci, K. Xu, R. Goroshin, C. Gelada, K. Swersky, P. Manzagol, H. Larochelle, Meta-dataset: A dataset of datasets for learning to learn from few examples, in: International Conference on Learning Representations, 2020.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: common objects in context, in: European conference on computer vision, Springer, 2014, pp. 740–755.

[33] J. Requeima, J. Gordon, J. Bronskill, S. Nowozin, R. E. Turner, Fast and flexible multi-task classification using conditional neural adaptive processes, in: Advances in Neural Information Processing Systems, 2019.

[34] P. Bateni, R. Goyal, V. Masrani, F. Wood, L. Sigal, Improved few-shot visual classification, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 14493–14502.

[35] W. Li, X. Liu, H. Bilen, Cross-domain few-shot learning with task-specific adapters, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.

[36] E. Xing, M. Jordan, S. J. Russell, A. Ng, Distance metric learning with application to clustering with side-information, Advances in Neural Information Processing Systems 15 (2002) 521–528.

[37] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, R. Shah, Signature verification using a "siamese" time delay neural network, International Journal of Pattern Recognition and Artificial Intelligence 7 (04) (1993) 669–688.

[38] H. Li, W. Dong, X. Mei, C. Ma, F. Huang, B.-G. Hu, LGM-Net: Learning to generate matching networks for few-shot learning, in: International Conference on Machine Learning, 2019, pp. 3825–3834.

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

[40] Y. Liu, J. Lee, M. Park, S. Kim, E. Yang, S. J. Hwang, Y. Yang, Learning to propagate labels: Transductive propagation network for few-shot learning, in: International Conference on Learning Representations, 2019.

[41] H.-Y. Tseng, H.-Y. Lee, J.-B. Huang, M.-H. Yang, Cross-domain few-shot classification via learned feature-wise transformation, in: International Conference on Learning Representations, 2020.

[42] W.-H. Li, X. Liu, H. Bilen, Universal representation learning from multiple domains for few-shot classification, in: IEEE/CVF International Conference on Computer Vision, 2021, pp. 9526–9535.

[43] W. Jiang, K. Huang, J. Geng, X. Deng, Multi-scale metric learning for few-shot learning, IEEE Transactions on Circuits and Systems for Video Technology (2020).

[44] Z. Wu, Y. Li, L. Guo, K. Jia, PARN: Position-aware relation networks for few-shot learning, in: IEEE International Conference on Computer Vision, 2019.

[45] W. Xu, Y. Xu, H. Wang, Z. Tu, Attentional constellation nets for few-shot learning, in: International Conference on Learning Representations, 2021.

[46] F. Wu, J. S. Smith, W. Lu, C. Pang, B. Zhang, Attentive prototype few-shot learning with capsule network-based embedding, in: European Conference on Computer Vision, Springer, 2020, pp. 237–253.

[47] H. Huang, J. Zhang, J. Zhang, J. Xu, Q. Wu, Low-rank pairwise alignment bilinear network for few-shot fine-grained image classification, IEEE Transactions on Multimedia (2020).

[48] C. Dong, W. Li, J. Huo, Z. Gu, Y. Gao, Learning task-aware local representations for few-shot learning, in: International Joint Conference on Artificial Intelligence, 2020.

[49] K. Cao, M. Brbic, J. Leskovec, Concept learners for few-shot learning, in: International Conference on Learning Representations, 2021.

[50] X.-S. Wei, P. Wang, L. Liu, C. Shen, J. Wu, Piecewise classifier mappings: Learning fine-grained learners for novel categories with few examples, IEEE Transactions on Image Processing 28 (12) (2019) 6116–6125.

[51] H. Huang, J. Zhang, L. Yu, J. Zhang, Q. Wu, C. Xu, TOAN: Target-oriented alignment network for fine-grained image categorization with few labeled samples, IEEE Transactions on Circuits and Systems for Video Technology (2021).

[52] J. Wu, T. Zhang, Y. Zhang, F. Wu, Task-aware part mining network for few-shot learning, in: IEEE/CVF International Conference on Computer Vision, 2021, pp. 8433–8442.

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

[54] S. W. Yoon, D.-Y. Kim, J. Seo, J. Moon, XtarNet: Learning to extract task-adaptive representation for incremental few-shot learning, in: International Conference on Machine Learning, 2020, pp. 10852–10860.

[55] S. Rahman, S. Khan, F. Porikli, A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning, IEEE Transactions on Image Processing 27 (11) (2018) 5652–5667.

[56] T. D. Kulkarni, W. F. Whitney, P. Kohli, J. Tenenbaum, Deep convolutional inverse graphics network, in: Advances in Neural Information Processing Systems, 2015, pp. 2539–2547.

[57] A. J. Ratner, H. Ehrenberg, Z. Hussain, J. Dunnmon, C. Ré, Learning to compose domain-specific transformations for data augmentation, in: Advances in Neural Information Processing Systems, 2017, pp. 3236–3246.

[58] L. Perez, J. Wang, The effectiveness of data augmentation in image classification using deep learning, arXiv preprint arXiv:1712.04621 (2017).

[59] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, R. Webb, Learning from simulated and unsupervised images through adversarial training, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2242–2251.

[60] A. Antoniou, A. Storkey, H. Edwards, Data augmentation generative adversarial networks, International Conference on Learning Representations Workshop (2018).

[61] A. J. Ratner, C. M. De Sa, S. Wu, D. Selsam, C. Ré, Data programming: Creating large training sets, quickly, in: Advances in Neural Information Processing Systems, 2016, pp. 3567–3575.

[62] Y.-X. Wang, R. Girshick, M. Hebert, B. Hariharan, Low-shot learning from imaginary data, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7278–7286.

[63] H. Zhang, J. Zhang, P. Koniusz, Few-shot learning via saliency-guided hallucination of samples, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2770–2779.

[64] J. Guan, Z. Lu, T. Xiang, A. Li, A. Zhao, J.-R. Wen, Zero and few shot learning with semantic feature synthesis and competitive learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (7) (2020) 2510–2523.

[65] S. Gidaris, A. Bursuc, N. Komodakis, P. Perez, M. Cord, Boosting few-shot visual learning with self-supervision, in: IEEE International Conference on Computer Vision, 2019.

[66] C. Liu, Y. Fu, C. Xu, S. Yang, J. Li, C. Wang, L. Zhang, Learning a few-shot embedding model with contrastive learning, in: AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 8635–8643.

[67] Y. Ouali, C. Hudelot, M. Tami, Spatial contrastive learning for few-shot classification, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2021, pp. 671–686.

[68] Z. Yang, J. Wang, Y. Zhu, Few-shot classification with contrastive learning, in: European Conference on Computer Vision, Springer, 2022, pp. 293–309.

[69] T. Xiao, X. Wang, A. A. Efros, T. Darrell, What should not be contrastive in contrastive learning, in: International Conference on Learning Representations, 2021.

[70] Y. Zhu, W. Min, S. Jiang, Attribute-guided feature learning for few-shot image recognition, IEEE Transactions on Multimedia 23 (2020) 1200–1209.

[71] L. Zhang, S. Wang, X. Chang, J. Liu, Z. Ge, Q. Zheng, Auto-FSL: Searching the attribute consistent network for few-shot learning, IEEE Transactions on Circuits and Systems for Video Technology (2021).

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

[73] W. Li, J. Xu, J. Huo, L. Wang, Y. Gao, J. Luo, Distribution consistency based covariance metric networks for few-shot learning, in: AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 8642–8649.

[74] K. Allen, E. Shelhamer, H. Shin, J. Tenenbaum, Infinite mixture prototypes for few-shot learning, in: International Conference on Machine Learning, 2019, pp. 232–241.

[75] C. Doersch, A. Gupta, A. Zisserman, CrossTransformers: spatially-aware few-shot transfer, in: Advances in Neural Information Processing Systems, 2020.

[76] J. Lu, S. Jin, J. Liang, C. Zhang, Robust few-shot learning for user-provided data, IEEE Transactions on Neural Networks and Learning Systems 32 (4) (2020) 1433–1447.

[77] C. Ma, Z. Huang, M. Gao, J. Xu, Few-shot learning via dirichlet tessellation ensemble, in: International Conference on Learning Representations, 2022.

[78] A. Ravichandran, R. Bhotika, S. Soatto, Few-shot learning with embedded class models and shot-free meta training, in: IEEE International Conference on Computer Vision, 2019.

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

[80] D. Das, C. G. Lee, A two-stage approach to few-shot learning for image recognition, IEEE Transactions on Image Processing 29 (2020) 3336–3350.

[81] S. W. Yoon, J. Seo, J. Moon, TapNet: Neural network augmented with task-adaptive projection for few-shot learning, in: International Conference on Machine Learning, 2019, pp. 7115–7123.

[82] A. Li, T. Luo, T. Xiang, W. Huang, L. Wang, Few-shot learning with global class representations, in: IEEE International Conference on Computer Vision, 2019.

[83] R. Chen, T. Chen, X. Hui, H. Wu, G. Li, L. Lin, Knowledge graph transfer network for few-shot recognition, in: AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 10575–10582.

[84] T. Chen, L. Lin, X. Hui, R. Chen, H. Wu, Knowledge-guided multi-label few-shot learning for general image recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence (2020).

[85] X. Luo, L. Wei, L. Wen, J. Yang, L. Xie, Z. Xu, Q. Tian, Rectifying the

shortcut learning of background for few-shot learning, in: Advances in Neural Information Processing Systems, 2021, pp. 13073–13085.

[86] Y. Zhou, Y. Guo, S. Hao, R. Hong, Hierarchical prototype refinement with progressive inter-categorical discrimination maximization for few-shot learning, IEEE Transactions on Image Processing (2022).

[87] Z. Sun, J. Wu, X. Li, W. Yang, J.-H. Xue, Amortized bayesian prototype meta-learning: A new probabilistic meta-learning approach to few-shot image classification, in: International Conference on Artificial Intelligence and Statistics, 2021, pp. 1414–1422.

[88] J. Zhang, C. Zhao, B. Ni, M. Xu, X. Yang, Variational few-shot learning, in: IEEE International Conference on Computer Vision, 2019.

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

[90] B. Liu, Y. Cao, Y. Lin, Q. Li, Z. Zhang, M. Long, H. Hu, Negative margin matters: Understanding margin in few-shot classification, in: European Conference on Computer Vision, Springer, 2020, pp. 438–455.

[91] W. Zhu, W. Li, H. Liao, J. Luo, Temperature network for few-shot learning with distribution-aware large-margin metric, Pattern Recognition 112 (2021) 107797.

[92] J. Chen, L.-M. Zhan, X.-M. Wu, F.-l. Chung, Variational metric scaling for metric-based meta-learning, in: AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 3478–3485.

[93] L. Qiao, Y. Shi, J. Li, Y. Wang, T. Huang, Y. Tian, Transductive episodic-wise adaptive metric for few-shot learning, in: IEEE International Conference on Computer Vision, 2019.

[94] V. N. Nguyen, S. Løkse, K. Wickstrøm, M. Kampffmeyer, D. Roverso, R. Jenssen, SEN: A novel feature normalization dissimilarity measure for prototypical few-shot learning networks, in: European Conference on Computer Vision, Vol. 12368, Springer, 2020, pp. 118–134.

[95] Y. Zheng, D. K. Pal, M. Savvides, Ring loss: Convex feature normalization for face recognition, in: IEEE conference on computer vision and pattern recognition, 2018, pp. 5089–5097.

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

[97] J. Xie, F. Long, J. Lv, Q. Wang, P. Li, Joint distribution matters: Deep brownian distance covariance for few-shot classification, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.

[98] X. Li, J. Wu, Z. Sun, Z. Ma, J. Cao, J.-H. Xue, BSNet: Bi-similarity network for few-shot fine-grained image classification, IEEE Transactions on Image Processing 30 (2020) 1318–1331.

[99] H. Zhang, P. Koniusz, S. Jian, H. Li, P. H. Torr, Rethinking class relations: Absolute-relative supervised and unsupervised few-shot learning, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 9432–9441.

[100] F. Hao, F. He, J. Cheng, L. Wang, J. Cao, D. Tao, Collect and select: Semantic alignment metric learning for few-shot learning, in: IEEE International Conference on Computer Vision, 2019.

[101] J. Kim, T. Kim, S. Kim, C. D. Yoo, Edge-labeling graph neural network for few-shot learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11–20.

[102] V. Garcia, J. Bruna, Few-shot learning with graph neural networks, International Conference on Learning Representations (2018).

[103] Y. Ma, S. Bai, S. An, W. Liu, A. Liu, X. Zhen, X. Liu, Transductive relation-propagation network for few-shot learning, in: International Joint Conference on Artificial Intelligence, 2020, pp. 804–810.

[104] C. Chen, K. Li, W. Wei, J. T. Zhou, Z. Zeng, Hierarchical graph neural networks for few-shot learning, IEEE Transactions on Circuits and Systems for Video Technology (2021).

[105] L. Yang, L. Li, Z. Zhang, X. Zhou, E. Zhou, Y. Liu, DPGN: Distribution propagation graph network for few-shot learning, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13390–13399.

[106] E. T. Oldewage, J. F. Bronskill, R. E. Turner, Attacking few-shot classifiers with adversarial support poisoning, in: ICML Workshop on Adversarial Machine Learning, 2021.

[107] M. Goldblum, L. Fowl, T. Goldstein, Adversarially robust few-shot learning: A meta-learning approach, Advances in Neural Information Processing Systems 33 (2020) 17886–17895.

[108] E. Bennequin, V. Bouvier, M. Tami, A. Toubhans, C. Hudelot, Bridging few-shot learning and adaptation: new challenges of support-query shift, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2021, pp. 554–569.

[109] M. Agarwal, M. Yurochkin, Y. Sun, On sensitivity of meta-learning to support data, Advances in Neural Information Processing Systems 34 (2021) 20447–20460.

[110] 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.

[111] M. McCloskey, N. J. Cohen, Catastrophic interference in connectionist networks: The sequential learning problem, in: Psychology of Learning and Motivation, Vol. 24, Elsevier, 1989, pp. 109–165.

[112] S. Gidaris, N. Komodakis, Dynamic few-shot visual learning without forgetting, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4367–4375.

[113] Y. Guo, N. C. Codella, L. Karlinsky, J. V. Codella, J. R. Smith, K. Saenko, T. Rosing, R. Feris, A broader study of cross-domain few-shot learning, in: European Conference on Computer Vision, Springer, 2020, pp. 124–141.

[114] K. Mahajan, M. Sharma, L. Vig, Meta-dermdiagnosis: Few-shot skin disease identification using meta-learning, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 730–731.

[115] X. Chen, L. Yao, T. Zhou, J. Dong, Y. Zhang, Momentum contrastive learning for few-shot covid-19 diagnosis from chest ct images, Pattern Recognition 113 (2021) 107826.

[116] D. Argüeso, A. Picon, U. Irusta, A. Medela, M. G. San-Emeterio, A. Bereciartua, A. Alvarez-Gila, Few-shot learning approach for plant disease classification using images taken in the field, Computers and Electronics in Agriculture 175 (2020) 105542.

[117] L. Li, X. Yao, G. Cheng, J. Han, Aifs-dataset for few-shot aerial image scene classification, IEEE Transactions on Geoscience and Remote Sensing 60 (2022) 1–11.

[118] X. Li, D. Shi, X. Diao, H. Xu, Scl-mlnet: Boosting few-shot remote sensing scene classification via self-supervised contrastive learning, IEEE Transactions on Geoscience and Remote Sensing 60 (2021) 1–12.

[119] Z. Li, M. Liu, Y. Chen, Y. Xu, W. Li, Q. Du, Deep cross-domain few-shot learning for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 60 (2021) 1–18.

[120] Z. Xue, Y. Zhou, P. Du, S3net: Spectral-spatial siamese network for few-shot hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing (2022).

[121] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, S. C. Hoi, Deep learning for person re-identification: A survey and outlook, IIEEE Transactions on Pattern Analysis and Machine Intelligence 44 (6) (2021) 2872–2893.

[122] G. Zou, G. Fu, X. Peng, Y. Liu, M. Gao, Z. Liu, Person re-identification based on metric learning: A survey, Multimedia Tools and Applications 80 (17) (2021) 26855–26888.

[123] W.-S. Zheng, S. Gong, T. Xiang, Towards open-world person re-identification by one-shot group-based verification, IEEE Transactions on Pattern Analysis and Machine Intelligence 38 (3) (2015) 591–606.

[124] Y. Wu, H. Liu, Y. Fu, Low-shot face recognition with hybrid classifiers, in: IEEE International Conference on Computer Vision Workshops, 2017, pp. 1933–1939.

[125] H. Du, H. Shi, Y. Liu, J. Wang, Z. Lei, D. Zeng, T. Mei, Semi-siamese training for shallow face learning, in: European Conference on Computer Vision, Springer, 2020, pp. 36–53.

[126] F. Cakir, K. He, X. Xia, B. Kulis, S. Sclaroff, Deep metric learning

to rank, in: IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 1861–1870.

[127] X. Shen, Y. Xiao, S. X. Hu, O. Sbai, M. Aubry, Re-ranking for image retrieval and transductive few-shot classification, Advances in Neural Information Processing Systems 34 (2021) 25932–25943.