International Journal of Machine Learning and Cybernetics (2022) 13:2495–2506
https://doi.org/10.1007/s13042-022-01539-1

ORIGINAL ARTICLE

Multi-distance metric network for few-shot learning

Farong Gao[1] · Lijie Cai[1] · Zhangyi Yang[1] · Shiji Song[2] · Cheng Wu[2]

Received: 3 June 2021 / Accepted: 5 March 2022 / Published online: 25 March 2022

Abstract

Few-shot learning aims to make classification when few samples are available. In general, metric-based methods map images into a space by learning the embedding function. However, conventional metric-based methods rely on a single distance value, which does not pay attention to the shallow features. In this paper, we propose a multi-distance metric network (MDM-Net) by employing a multi-output embedding network to map samples into different feature spaces. In addition, we maximize the inter-class distance which is popular in metric learning field to improve the performance of few-shot classifier. Furthermore, we design a task-adaptive margin to adjust the distance between different sample pairs, and we found that the distance loss combined with cross-entropy loss is beneficial to achieve better results in meta-task training. The proposed method is verified by tests on miniImageNet and FC100 these two benchmarks for 5-way 1-shot classification task and 5-way 5-shot classification task with competitive results.

Keywords Deep learning · Few-shot learning · Image recognition · Embedding space · Metric learning

1 Introduction

In the image recognition field, deep learning has the advantage of a powerful capability of feature extraction [1]. The AlexNet [2] won the first award on 2012 ImageNet [3] competition, defeating the traditional computer vision technology for the first time. From then on, many algorithms, such as GoogLeNet [4] and ResNet [5], have been proposed, which increased the complexity of the structure of convolutional neural networks (CNNs) but improved the recognition effect to a certain extent. Besides, in certain categories of recognition tasks, computer vision has surpassed humans. Therefore, deep learning has been becoming more and more popular to be applied in the vision-related tasks, such as object detection [5–9] and intelligent driving [10], and it has also achieved great progress in these areas. However, the powerful recognition effect of deep learning is obtained through repeated iterative training using a large amount of labeled data, namely, when the amount of training data is insufficient, the traditional CNNs can difficultly optimize millions of network parameters [11], and overfitting can be easily caused; thus, it can be unfeasible to achieve effective recognition [12–14].

Data augmentation [15] is a method to deal with overfitting. This method has been widely used in image recognition tasks when there is the insufficient number of samples [16], and the data have been expanded by transforming the original image in different processes, such as noise addition and rotation, flipping, scaling, cropping, and translation of an image [17]. However, the effects of such processes are limited because new data cannot be obtained indefinitely. Thus, data augmentation can reduce overfitting only to a certain extent when there are few numbers of samples.

Humans can easily identify new data based on a small number of labeled images, and can even identify a new class based on simple descriptions, which are similar to zero-shot learning [18–20]. However, this process can hardly be performed by machines. Generally, there are many relationships between different samples, and the knowledge obtained from the previous samples can be useful for understanding the current samples. In the theory of transfer learning, it is to imitate human and apply the knowledge learned from source domain to the target domain recognition of a novel category can be achieved by freezing the shallow network parameters and fine-tuning the deep network layers using a small

✉ Farong Gao
frgao@hdu.edu.cn

[1] School of Automation, Hangzhou Dianzi University, Hangzhou 310018, China

[2] Department of Automation, Tsinghua University, Beijing 100084, China

number of samples [21–23]. However, a pre-trained network is still required to perform supervised learning with a large amount of samples, and a certain amount of labeled samples are needed to fine-tune the network [24]. If the new samples in the target domain are too different in features from the samples in the source domain, the recognition effect will be affected greatly [25].

In order to overcome the problem of large samples amounts needed for model training, the image recognition methods by using a fewer number of samples have been widely developed. For example, Lake et al. [26] proposed a hierarchical Bayesian model based on composition and causality, which can learn various natural visual concepts from an image and summarize them in a human way. Xie et al. [27] proposed a method of using the knowledge graph of entity description to complete the entity classification under the zero-sample setting. Triantafillou et al. [28] designed a few shot visual learning system. During the test, it can effectively learn new categories from only a few training data without forgetting the initial categories to train. In the current study, there are three types of methods for few-shot learning, i.e., the model-based methods, the optimization-based methods, and the metric-based methods [29–31]. Santoro et al. [29] used a memory enhancement method to realize few-shot learning. Ravi et al. [30] presented the long short-term memory (LSTM) to achieve parameter optimization while considering the short-term and long-term information, which is more suitable for few-shot tasks, compared to the conventional gradient optimization algorithm. On the other hand, when non-parametric optimization methods such as the support vector machine (SVM), k-nearest neighbors (kNN) and other machine learning methods are implemented, the overfitting phenomena [32] can be alleviated effectively [33].

Metric-based methods [34] can classify the novel class when only few labeled samples are available. This type of method uses *M*-way *m*-shot setting (meta-task) to train the model, that is, randomly selects *M* classes from the dataset, and then selects m labeled samples as the training samples from the selected classes. In this way, each episode will contain different combinations of sample pairs, which is beneficial to improve the model generalization ability. However, metric-based methods usually learn only one embedding space. That is, the feature information is represented in only one feature space, and sample is not fully utilized, which is inappropriate in few-shot learning tasks. When the features of different classes are similar, for example, there are different breeds of dogs in miniImageNet [35], so the judgments made based on only a single distance are unreliable. Therefore, when comparing the similarity of the similar categories, the local features need to be taken more attention. On the other hand, Triplet loss [36] is a popular loss function in the field of metric learning, which is to make the distance from a sample to a negative sample larger than the distance to a positive sample by setting a margin. However the margin is set to be a fixed value. Namely, the minimum difference of distance between any positive sample pairs and any negative sample pairs is same and fixed.

The major work of this paper is as follows. (1) A multi-distance metric network (MDM-Net) is proposed for few-shot learning task to embed samples into different spaces. (2) Multi-level feature maps are considered in our method, because the shallow network layers contain more detail information, while the deep layers focus more on the abstract features of the image. (3) Introduce feature transformation into meta-task training which is beneficial to improve model performance. (4) Introduce inter-class distance into loss function with cross entropy together and propose a dynamic margin between different sample pairs. Finally, the proposed network is verified by tests with miniImageNet [35] and FC100 [37] benchmarks.

The rest of the paper is organized as following. In Sect. 2 is introduced. In Sect. 3, the specific method of the MDM-Net will be proposed. In Sect. 4, the experiments will be implemented to verify the proposed network, and in Sect. 5, there is the conclusion of this paper.

2 Related work

Metric-based few shot learning method does not make classification directly according to the extracted features like traditional CNN, but compute the similarity with the labeled samples by distance metric. Specifically, the image can be mapped into a feature space by learning an embedding function. Then the distance can be computed by comparing two different feature vectors. Thereby, the similarity between two samples can be expressed by this distance. Most of the recently developed outstanding models for few-shot learning have been based on this method. Koch et al. [38] proposed the Siamese network algorithm. They design a dual-loop neural network and input different combinations of sample pairs to the network. Vinyals et al. [35] proposed the matching network that constructs different encoders and uses a cosine distance to compute the similarity between samples, then the weighted summation is used to classify. Snell et al. [39] proposed the prototypical networks (P-Nets) algorithm. They obtain the prototype of each class by averaging the feature vectors of the same class. The prototype represents the center of the distribution of the samples which belong to the same class. When new sample is input, the Euclidean distances from this new sample to each prototype are able to be computed, the shorter the distance is, the greater the similarity and the probability of belonging to this class. Oreshkin [37] proposed task dependent adaptive metric for improved few-shot classification task by learning a task-dependent metric space. Li et al. [40] proposed a local descriptor based image-to-class measure to classify because a measure at such a level may not be effective enough. The prior few shot learning methods mostly used four-layer convolutional neural network as backbone, but now deep convolutional networks such as ResNet12 are more and more popular to be employed as feature extractor after pre-training which improve the performance of the classifier greatly. Li et al. [41] proposed an asymmetric distribution measure that is more suitable for metric-based few shot learning. Considering that it is not comprehensive and effective to calculate the similarity between different categories only from a single level, Chen et al. [42] proposed a multi-level metric learning (MML) method which used pixel-level features combined with global-level features and part-level features to improve the classification accuracy.

3 Multi-distance metric networks

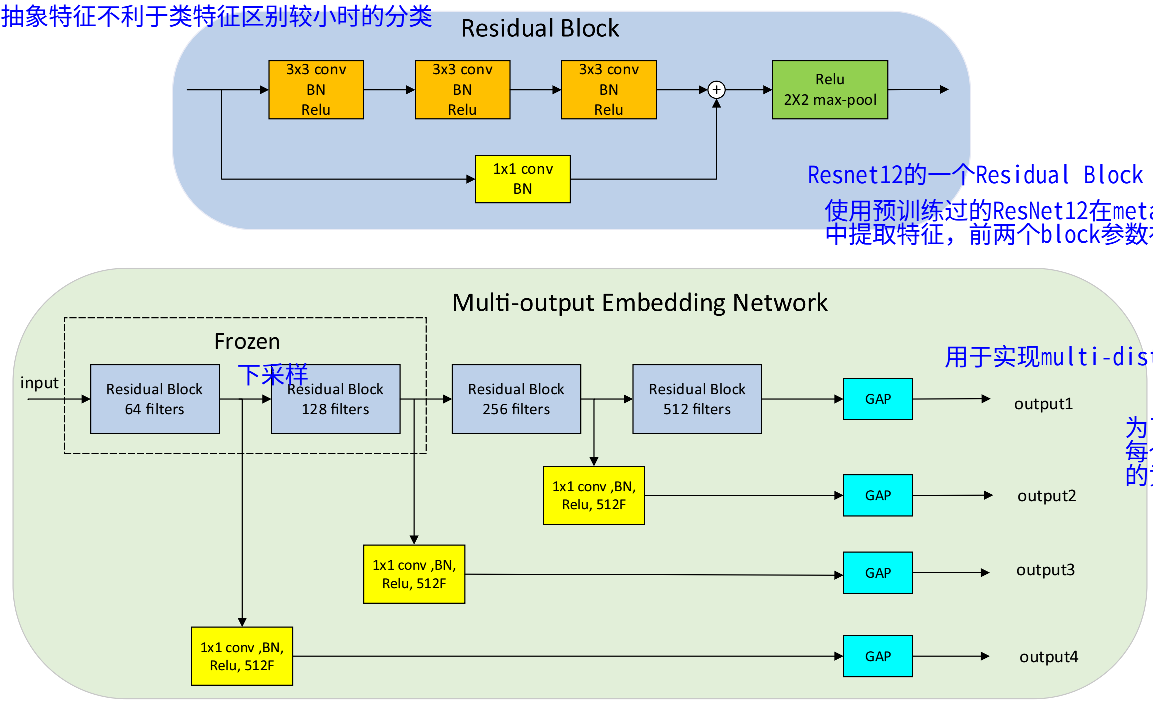
The features extracted from the ordinary network structure only retain the deep abstract features. This is not helpful when faced with classification tasks with small differences in category features. Therefore, we will use the shallow features to measure the distance to construct a distance that contains multiple feature layers. Few shot learning classification tasks are different from fine-grained image recognition tasks. The fixed distance value as a parameter of the distance loss function is not appropriate for scenarios with large variations in category features. Therefore, we designed an adaptive parameter to adjust the distance loss under different training tasks.

3.1 Multi-output embedding network architecture

The metric-based few-shot learning algorithms convert images into an embedding vector by employing CNN called the embedding network. Recently, many excellent methods employ ResNet12 as backbone instead of a four-layer convolutional neural network which used to be popular. We also adopt ResNet12 to perform pre-training on the training data for conventional 64-way classification task. Then, we remove the fully connected layers and freeze the parameters of the first two blocks. We randomly sample meta-task to fine-tune this network. Moreover, in the meta-task training phase, in order to make full use of multi-level feature maps, we propose a multi-output embedding network as shown in Fig. 1.

The proposed multi-output embedding network architecture is presented in Fig. 1. The ResNet12 is composed of four residual blocks, each block contain 3 convolutional layers. The pre-trained ResNet12 is used to extract features during the meta-task training phase. And the parameters of the first two blocks are frozen. GAP means global average pooling. This network is designed based mainly on the idea of multi-level feature fusion. The feature maps represent the

只保留深度抽象特征不利于类特征区别较小时的分类



Resnet12的一个Residual Block

使用预训练过的ResNet12在meta-task training中提取特征，前两个block参数被冻结

用于实现multi-distance metric

为了保证准确率，每个嵌入空间有相同的贡献：维度相同

深层特征：抽象信息
浅层特征：图像局部信息
不同类的图像比较相似时，可以关注浅层特征

Fig. 1 Multi-output embedding network

information of the images which the network learned. The deep-level features which learned by the deep layers contain the abstract information, and as for the shallow layers, the network focus on the local information of the image. In the few-shot classification tasks, sometimes, the images of different classes are similar, for example, there are different categories belong to the same superclass, so they have similar abstract feature, therefore we should pay more attention to the details.

使得样本可以从多层级角度分类

An extra channel is derived from each residual block, so that the sample can be classify from multi-level perspective. We use multi-level feature maps to achieve multi-distance metric, and in order to ensure the accuracy of classification, each embedding space should have the same level contribution to the classifier. Namely, different embedding spaces should keep the same dimension. We set a 1 × 1 convolutional layer with 512 filters to expand the output of each block to the same number of channels and add the output of the previous block to the next block's after down sampling. Finally four 512 dimension embedding vectors can be obtained per sample after a global average pooling processing. The vectors represent the positions of samples in the embedding space, and the distances between different samples are computed by using these vectors and used to represent the similarity between samples. In the multi-distance metric network, the samples are mapped into different spaces, and multiple distance values are obtained by computing the embedding vectors in these spaces.

3.2 Multi-distance metric method

The embedding vectors of the support samples are used to compute the prototype, and the distance between query sample and the prototype is computed in each embedding space. In the prediction phase, the distance values in different spaces are superimposed. In an embedding space, samples of the same class are close, while samples of different classes are far away. If there is only one embedding space, this space represents the deep abstract feature space. For some categories with small feature differences, their distribution in such space is very close, although they do not belong to the same category, which is not conducive to classification. In this case, the prediction based on the single distance is not accurate. Therefore, a multi-distance based method is proposed in this work. Namely, more information can be obtained by computing from multi-level feature maps. The distance values computed by different information obtained from different embedding spaces are inconsistent but related. These distance values from all the embedding spaces are combined to enlarge the feature difference and improve the classification accuracy. The shorter the distance from the query sample to the prototype, the greater the probability it is that the sample belong to the class represented by this prototype. The framework of the proposed multi-distance metric network is presented in Fig. 2.

预测阶段，不同空间中的距离值会被叠加

一些类别的特征间区别比较小，不利于分类
只根据单个特征空间得到分类结果不够准确。

As shown in Fig. 2, the support samples are used to compute the prototypes (the white points) for each class in different embedding spaces. All the samples computed by the embedding network are distributed in the embedding spaces in the form of points. By inputting the query sample (the yellow point), the corresponding query embedding vectors are obtained. Then the distance from query sample to each prototype in each space can be computed. For example, in the 3-way 3-shot training task, we can obtain a total of 12 distance values for each query sample.

Before computing the distance, we adopt feature transformation which proposed by [43] to normalize the embedding vectors, but since we use meta-task method for training which different from the original method, therefore we design a different center processing to replace which is suitable for meta-task.

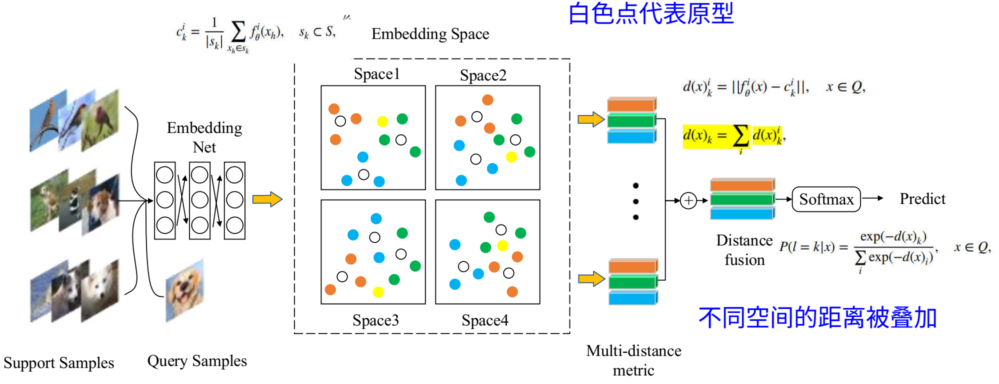
$$f^i_\theta(x) \leftarrow f^i_\theta(x) - \frac{1}{|X|}\sum_{x\in X} f^i_\theta(x), X \subset \{S, Q\}, \tag{1}$$

feature transformation

$$f^i_\theta(x) \leftarrow \frac{f^i_\theta(x)}{||f^i_\theta(x)||_2}, \tag{2}$$

where i represents different embedding spaces and also means ith-level feature maps, f represents the

Fig. 2 The framework of the proposed multi-distance metric network


$c^i_k = \frac{1}{|s_k|}\sum_{x_h\in s_k} f^i_\theta(x_h), \quad s_k \subset S,$
Embedding Space
Space1
Space2
Space3
Space4
Embedding Net
Support Samples
Query Samples
$d(x)^i_k = ||f^i_\theta(x) - c^i_k||, \quad x \in Q,$
$d(x)_k = \sum_i d(x)^i_k,$
Softmax
Predict
Distance fusion
$P(l = k|x) = \frac{\exp(-d(x)_k)}{\sum_i \exp(-d(x)_i)}, \quad x \in Q,$
Multi-distance metric

白色点代表原型

不同空间的距离被叠加

如果只有一个类别空间，当一些类别的特征间
区别较小，分类可能不准确

embedding function and $|X|$ represents the number of samples in the dataset X. S, Q represent support set and query set, respectively.

During meta-task training phase, we randomly select M classes from the train dataset for each episode, and then select N labeled samples as support set from these selected classes. And we also chose n samples from the same classes as query set. Then we need to compute the prototype for each class which represents the class center by averaging the embedding vectors. The prototypes of each class are as follows,

为每个类计算原型

原型，类中心

$$c_k^i = \frac{1}{|s_k|} \sum_{x_h \in s_k} f_\theta^i(x_h), \quad s_k \subset S, \tag{3}$$

where s_k denotes kth class of samples in the support set. In the fusion phase, the distance values of the same query sample are superimposed. The mathematical expression of distance is

$$d(x)_k^i = ||f_\theta^i(x) - c_k^i||, \quad x \in Q, \tag{4}$$

$$d(x)_k = \sum_i d(x)_k^i, \tag{5}$$

where $f_\theta^i(x)$ denotes the query set sample in the ith embedding space, c_k^i denotes the prototype of the same category sample in the i th embedding space. In addition, $d(x)_k$ need to be normalized (Min–Max Normalization). Then, we use softmax to compute the probability of belonging to each class,

$$P(l = k|x) = \frac{\exp(-d(x)_k)}{\sum_i \exp(-d(x)_i)}, \quad x \in Q, \tag{6}$$

where k represents the label of the query sample.

Compared to the other metric-based few-shot learning methods, for the proposed MDM-Net, the number of embedding spaces is increased, and distance in each space is computed to improve the classification accuracy. In addition, we continue to consider more distance. We set image transformation operating before seeding it into the network. During each episode of training and testing, a sample is flipped horizontally and then input to the network with the original sample simultaneously. Thus, we sharpen the two different predicted results [44]. The specific operation processes are shown in Fig. 3.

As shown in Fig. 3, the original image and flipped image are converted into two vectors, in which p_1 and p_2, respectively, represent the predicted results of the original sample and flipped sample. Then the two parts are averaged and sharpened, while the features of the sample are not changed after being flipped. The mathematical expressions of sharpen operations are as follows,

$$x^F = Flipped(x), \tag{7}$$

$$p(l = k|x)_{Avg} = Avg(p(l = k|x), p(l = k|x^F)), \tag{8}$$

$$Sharpen(x, T)_k = p(l = k|x)_{Avg}^{\frac{1}{T}} \Big/ \sum_j p(l = j|x)_{Avg}^{\frac{1}{T}}, \tag{9}$$

We set T to 0.5 and use cross entropy to calculate the classification loss,

$$loss_c = -\sum_{x \in Q} \log(Sharpen(x, T)_k). \tag{10}$$

However, the cross-entropy loss function only focuses on positive samples and weakens the effect of negative samples during training. We introduce distance loss and increase the contribution of negative samples to parameter updates. By adjusting the distance between different samples, the model can learn more subtle feature information. Therefore, we propose a loss function of cross entropy loss combined with distance loss to improve the classification ability of the model.

3.3 Distance loss

triplet loss的原理?

损失函数公式：$L = max(d(a,p) - d(a,n) + margin, 0)$

Triplet loss only compute one positive sample pair and one negative sample pair at a time, and we extend it to the M-way N-shot classification task, which simultaneously compute a positive sample pair and M-1 negative samples pairs. Therefore, the mathematical expression of distance loss is,

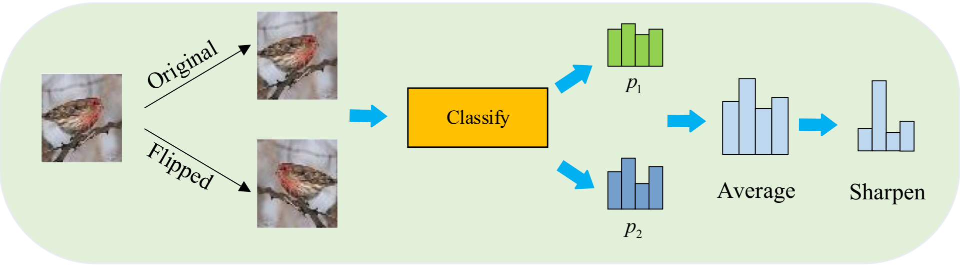


Fig. 3 Each image is flipped before being embedded

任务自适应去找到一个合适的m

$$loss_d = \sum_{x \in Q} \sum_{j \neq k} \sum_{i} \max(d(x)_k^i - d(x)_j^i + m, 0), \tag{11}$$

where k denotes the label of query sample, and m is a hyper parameter. In metric learning, m is set to be a fixed value and denotes a margin. We improve the setting of m, since each meta-learning task is randomly sampled, which includes great amount combinations of sample pairs, this paper proposes a task-adaptive method for different meta-tasks to compute a suitable margin.

The adaptive method is reflected in the model training stage, which is conducive to model convergence. Because the distance loss function is introduced in metric learning, and the feature difference between different sample pairs in the data set changes greatly.

Before each meta-task training, we perform task adaptation processing on support set, that is, compute the similarity among each class before training. If two different classes are similar in features, the margin (m) should be reduced. And if two classes are different in features, the margin should be increased. Moreover, as the similarity decreases, the trend of increasing m needs to be gradually slowed down, because the margin is big enough to assist model to make classification. What is more, a larger distance difference will affect the optimization of the model. We should pay more attention to the categories with high similarity, because the model is still difficult to identify them. Therefore, we use the logarithmic method to compute the value of m.

$$z_{hj}^i = ||f_\theta^i(x_h) - f_\theta^i(x_j)||^2, x_h, x_j \in S, N = 1, \tag{12}$$

$$(z_h^i)_{\min} = \min\{z_{hj}^i | j \neq h\}, \tag{13}$$

$$TA(m)_{hj}^i = (1 + \log(1 + z_{hj}^i - (z_h^i)_{\min})) * m, j \neq h, \tag{14}$$

where z_{hj}^i represents the distance matrix between samples x_h and x_j of different classes, k represents the label of the sample, and N represents the number of support sample per class. Obviously, $TA(m)_{\min} = m$, namely, m is the shortest distance between a sample and another sample. In addition, in few-shot classification tasks ($N > 1$), we use prototypes to compute the value of m.

$$z_{hj}^i = ||c_h^i - c_j^i||^2, N > 1, \tag{15}$$

where z_{hj}^i represents the distance matrix between prototypes c_h^i and c_j^i of different classes. Therefore the mathematical expression of the distance loss is,

$$loss_d = \sum_{x \in Q} \sum_{j \neq k} \sum_{i} \max(d(x)_k^i - d(x)_j^i + TA(m)_j^i, 0), \tag{16}$$

the final loss function is combined with cross-entropy loss and distance loss,

$$loss = loss_c + \alpha * loss_d, \tag{17}$$

where α represents the scale factor, and we set it to 0.2.

4 Experiments

We implement experiments on the miniImageNet datasets [42] and the CIFAR100 datasets [45] to verify the proposed method in this paper. The performance of the MDM-Net was compared with those related previous works on these two few-shot learning benchmarks. All experiments were performed on a GTX 1080Ti GPU using the TensorFlow deep learning framework. We perform 5-way 5-shot classification task and 5-way 1-shot classification task to evaluate our model according to the popular experiment setting of few-shot learning. Specifically, five classes were randomly selected from the novel classes, and 1(5) samples from every selected class were selected as support samples. Meanwhile, 15 query samples per class were also selected. Because the P-Net is popular in the field of few shot learning and widely used in later works. The method proposed in this paper also use the idea of prototype, therefore the P-Nets with ResNet can still be a baseline model. We propose multi-distance metric method based on P-Nets and perform feature transformation as well as distance loss to improve the classification ability of the baseline model.

4.1 Datasets

The miniImageNet [35] denotes a dataset proposed for few-shot learning tasks, and it has been widely used in performance evaluating of few-shot learning methods. This popular few-shot learning benchmark is proposed by [35], and all the samples were derived from the ILSVRC-2012 dataset. The miniImageNet dataset contains 100 classes, and each class consists of 600 RGB images, so there were a total of 60,000 images. We follow [35] to split classes into 64 training classes, 16 validation classes and 20 testing classes.

The FC100 dataset is a few shot learning version of CIFAR100 which include 100 classes with a total of 60,000 images, where every class contained 600 images with a size of 32×32 pixels, what is more, these 100 classes are grouped into 20 super classes. We follow [37] to split classes into 60 training classes, 20 validation classes and 20 testing classes. Compared to the miniImageNet, the images in FC100 are smaller in size and different classes in the same superclass are similar which make the classification task more difficult. Therefore, it can also be considered as an experimental dataset for the few-shot learning methods in this paper.

4.2 Implementation details

In the few-shot learning task on the miniImageNet dataset, at first, all the images were converted to a uniform size of 84 × 84 pixels according to the common data division principle of this benchmark. We employ the output of the last two blocks as feature vectors ($i \in [3, 4]$) because the samples in miniImageNet have abundant background information which would affect the classification. Differently, in the few-shot learning task on FC100 dataset, we employ the output of all the blocks as feature vectors ($i \in [1, 4][1, 4]$).Before training, we flip all the samples in the training set vertically. The initial learning rate was set to 0.001, which dropped by 20% after every 2000 episodes. We test repeated 600 times to compute the average accuracy and give a 95% confidence interval.

4.3 Results

Table 1 shows the influence of feature transformation (FT) on the results, where **Y** denotes the feature transformation is adopted and **N** is not. MDM-Net represents multi-distance metric network which proposed in this article. We adopt the settings of 5-way 1-shot and 5-way 5-shot on miniImagenet and FC100 to verify the effectiveness of our method.

The feature transformation improves the classification ability of the model for 1-shot classification task. As shown in Table 1, we improve the result from 58.05 to 59.88% on miniImageNet and improve the result from 42.50 to 43.62% on FC100 for 5-way 1-shot classification task. Obviously, normalizing feature vectors can still improve model performance when perform meta-tasks training for 1-shot tasks. However, on the other hand, feature transformation has no effect on 5-shot tasks, and even the accuracy has dropped. Therefore we propose to make feature transformation in the case of 1-shot.

The few-shot learning algorithm based on the prototype network greatly is dependent on the reliability of the prototype in terms of recognition effect. The prototype is the center of a category in space. In the 1-shot experiment, because there is just one sample and the classification center cannot be calculated, the prototype is unreliable and the model is difficult to converge. However, the space can be reduced and the error can be decreased by the feature transformation. In the 5-shot experiment, the prototype is relatively reliable, so there is no need for feature transformation, because the feature transformation itself changes the distribution of samples in the embedding space.

The results of different settings perform on miniImageNet dataset and FC100 dataset are shown in Table 2, we also adopt the settings of 5-way 5-shot and 5-way 1-shot classification tasks to test our method for few shot learning task on different datasets. Especially, **Y** denotes that we adopt the corresponding methods (multi-distance metric network, image flipped or an adaptive margin), and **N** means not.

Obviously, when we only use the multi-distance metric method, the recognition effect of the model is improved significantly. In addition, when we adopt the multi-distance metric method with the adaptive distance loss function together, the recognition effect of the model has been greatly improved.

Figure 4 shows different feature maps learned by multi-level layers, in which (a) and (c) are the original image and the flipped image, (b) and (d) are the output images corresponding to the shallow network, (c), (d), (g), (h) respectively represent the output images of four different dimensional channels of the original image from low dimension to high dimension. These feature maps are produced by

Table 1 The influence of feature transformation (FT) to the results

Model	FT	miniImageNet		FC100	
		5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
MDM-Net	N	58.08 ± 0.55	**76.60 ± 0.24**	42.50 ± 0.28	**57.41 ± 0.33**
MDM-Net	Y	**59.88 ± 0.42**	75.45 ± 0.34	**43.62 ± 0.46**	56.45 ± 0.25

Table 2 The influence of multi-distance and adaptive margin on the results

Model	Image flipped	Multi-distance	m-adaptive	miniImageNet		FC100	
				5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
MDM-Net	Y	N	N	56.11 ± 0.55	73.30 ± 0.43	37.79 ± 0.35	52.57 ± 0.18
MDM-Net	Y	N	Y	56.52 ± 0.50	74.15 ± 0.18	38.42 ± 0.36	53.53 ± 0.25
MDM-Net	Y	Y	N	57.70 ± 0.26	75.50 ± 0.25	41.51 ± 0.52	55.83 ± 0.21
MDM-Net	Y	Y	Y	59.88 ± 0.42	76.60 ± 0.24	43.62 ± 0.46	57.41 ± 0.33

different blocks. For the few shot learning task on miniImageNet, we use the last two level feature maps to make classification. We adopt the form of multiple outputs and image transformation to get multiple distances. Although the image transformation is similar to data augmentation, the difference is that data augment processing only increases the number of training samples, and the method proposed in this paper focuses more on the relationship between transformed images and the original images when they are embedded. The network learns these relationships through distance computing and comparing with the similarity among different support samples or prototypes. Moreover multi-level feature maps contain more information which is beneficial for few-shot learning task. Obviously, the features learned by the deep-level layers are abstract. But when different classes are similar, we propose to pay attention to the local features. However, when the background information of the image is complicated, there will be a lot of interference information in the shallow-level feature maps, and it is necessary to choose different level feature maps reasonably for different dataset. Therefore we chose the last two blocks' output as feature vectors without feature fusion when evaluating our method on miniImageNet. The experimental results show that our method is effective. Apparently, the feature vectors we obtained contain the feature information of different residual blocks, the information loss was reduced during the network transmission, and the feature extraction capability of the network was also stronger than the original network.

As shown in Fig. 5, the MDM represents multi-distance metric method, and dloss represents distance loss which is task-adaptive. We achieve 52.66% accuracy for 5-way 1-shot task by only employing a pre-trained backbone without meta-training. After multi-distance metric method combines with distance loss, the result of classification is improved faster and achieving a better accuracy which shows that distance loss effectively improves the classification ability of the model. We improving the classification accuracy from 57.02 to 59.95% for 5-way 1-shot learning task on miniImageNet and proves that distance loss is beneficial to meta training. Since the distance between samples of different categories (inter-class distance) is enlarged by employing distance loss function, and the margins between samples with similar features are larger than those with different features', which is more conducive to improving the classification ability of the model.

We set m which is task-adaptive to adjust the minimum distance between different categories. When the categories in the meta-task are similar, the value of m will be reduced appropriately, namely, the inter-class distance will be shorter. According to experimental results, a suitable inter-class distance can improve the performance of few shot classifier. We test the influence of different margins on the classification results. As shown in Figs. 6, 7, 8, 9, for 1-shot classification task, we obtain the best validation result when setting m to 0.15 on both FC100 and miniImageNet. As for

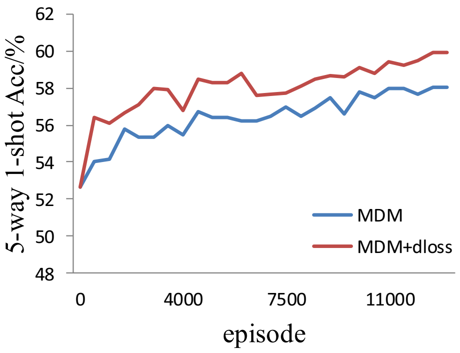
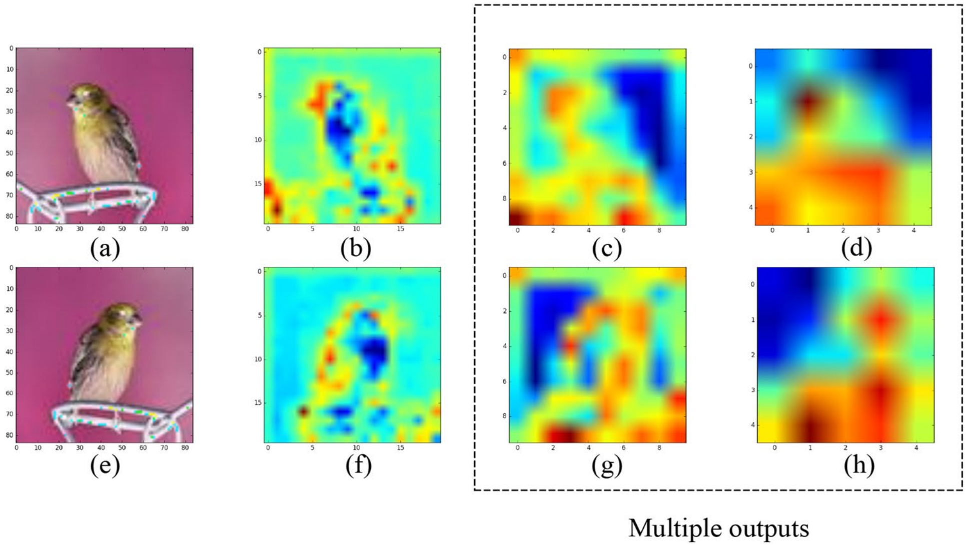


Fig. 5 The accuracy of 5way-1shot classification task on miniImageNet

Fig. 4 Multi-level feature maps of the original image and the flipped image



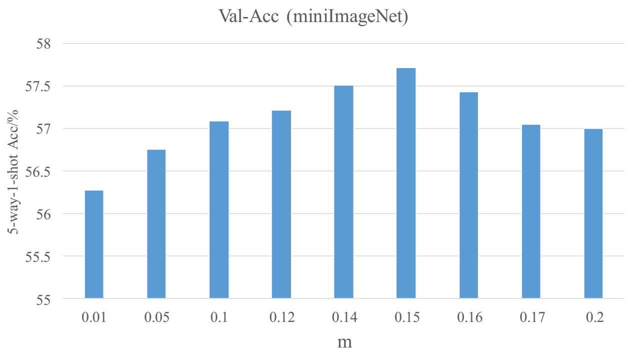


Fig. 6 The 5-way 1-shot classification accuracy on miniImageNet

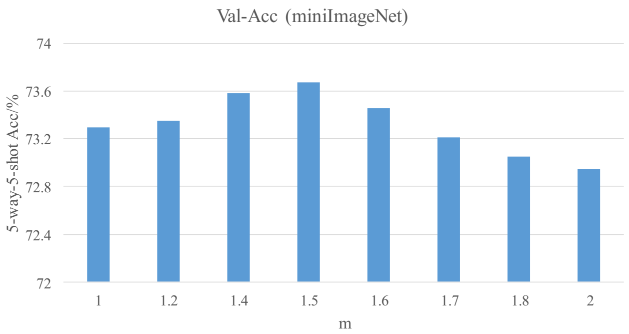


Fig. 8 The 5-way 5-shot classification accuracy on miniImageNet

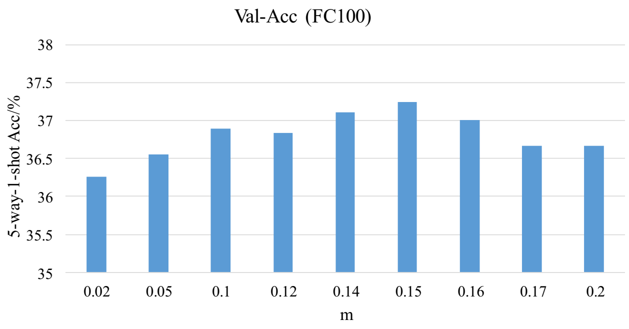


Fig. 7 The 5-way 1-shot classification accuracy on FC100

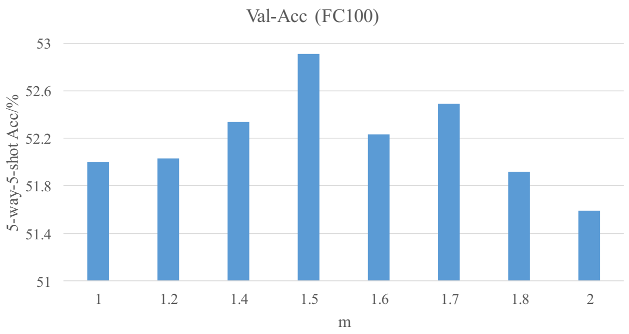


Fig. 9 The 5-way 5-shot classification accuracy on FC100

5-shot classification task, we achieve the best validation result when setting m to 1.5 (Table 3). Because we did not normalize the feature vectors when deal with 5-shot classification task, therefore the distance value between the samples were larger than 1-shot's, so that we need to set a larger margin. We evaluate our model on test set with obtained margins. As shown in Table 4 and Table 5, we achieved similar results to the validation set.

The results of different methods perform on miniImageNet dataset and FC100 dataset are shown as follows**.** We adopt 5-way 5-shot and 5-way 1-shot to test our method for few shot learning task on different data set and compare with existing algorithms.

Compared with the result of 5-shot classification task, the accuracy of 1-shot classification task was significantly improved. Since the previous metric-based method used the prototype to represent each class, the average value could not be obtained in 1-shot classification task because there was only single support sample per class. Therefore, the ability of the prototype to represent classes was obviously insufficient.

However, our MDM-Net could obtain multiple distance values for prediction to improve the result even when there was only one sample. Comparing with our baseline model (P-Net+ResNet), we improve the accuracy from 54.16 to 59.88% for 5-way 1-shot classification task on miniImageNet and improve the accuracy from 73.68 to 76.60% for 5-way 5-shot classification task. In addition, we improve the result from 37.80 to 43.62% for 1-shot task on FC100 and improve the accuracy from 55.30 to 57.41% for 5-shot task on the same benchmark which are competitive compared to the state of the art model. Obviously, compared with other methods, our method performs better on the FC100 dataset, because the features between the categories in FC100 are similar, and the multi-distance measurement method can help the model learn more feature information. The test set

Table 3 The influence of different margins to the classification results on test set

Model	Margin	miniImageNet	FC100
5-way 1-shot (%)			
MDM-Net	m = 0.05	58.55 ± 0.25	42.25 ± 0.31
MDM-Net	m = 0.15	**59.88 ± 0.42**	**43.62 ± 0.46**
MDM-Net	m = 0.5	58.26 ± 0.43	41.90 ± 0.43
5-way 5-shot (%)			
MDM-Net	m = 1.0	74.34 ± 0.36	57.11 ± 0.23
MDM-Net	m = 1.5	**76.60 ± 0.24**	**57.41 ± 0.33**
MDM-Net	m = 2.0	75.66 ± 0.26	56.50 ± 0.28

Table 4 The result of few-shot learning on miniImageNet dataset

Model	Backbone	5-way 1-shot (%)	5-way 5-shot (%)
Meta LSTM [46]	Conv-4	43.44 ± 0.77	60.60 ± 0.71
Matching networks [35]	Conv-4	43.40 ± 0.78	51.09 ± 0.71
Matching networks FCE [35]	Conv-4	43.56 ± 0.84	55.31 ± 0.73
MAML [47]	Conv-4	48.70 ± 1.84	63.15 ± 0.91
P-Net [39]	Conv-4	49.42 ± 0.78	68.20 ± 0.66
PLATIPUS [48]	Conv-4	50.13 ± 1.86	–
mAP-SSVM [28]	Conv-4	50.32 ± 0.80	63.94 ± 0.72
GNN [49]	Conv-4	50.33 ± 0.36	66.41 ± 0.63
Relation Net [50]	Conv-4	50.44 ± 0.82	65.32 ± 0.70
MTNet [51]	Conv-4	51.70 ± 1.84	–
MAML [47]	ResNet-18	49.61 ± 0.92	65.72 ± 0.77
RelationNet [50]	ResNet-18	52.48 ± 0.86	69.83 ± 0.68
MatchingNet [35]	ResNet-18	52.91 ± 0.88	68.88 ± 0.69
P-Net [39]	ResNet-18	54.16 ± 0.82	73.68 ± 0.65
SNAIL [52]	ResNet-15	55.71 ± 0.99	68.88 ± 0.92
adaCNN [53]	ResNet-15	56.88 ± 0.62	71.94 ± 0.57
TADAM [37]	ResNet-12	58.50 ± 0.30	**76.70 ± 0.30**
TPN [54]	ResNet-12	59.46	73.74
MTL [55]	ResNet-12	**61.2 ± 1.8**	75.5 ± 0.8
MDM-Net (Ours)	ResNet-12	59.88 ± 0.42	76.60 ± 0.24

Table 5 The result of few-shot learning on FC100 dataset

Model	Backbone	5-way 1-shot (%)	5-way 5-shot (%)
P-Net [39]	ResNet-12	37.80 ± 0.40	53.30 ± 0.50
Cosine classifier [56]	ResNet-12	38.47 ± 0.70	**57.67 ± 0.77**
TADAM [37]	ResNet-12	40.10 ± 0.40	56.10 ± 0.40
SimpleShot [43]	ResNet-10	40.13 ± 0.18	53.63 ± 0.18
Metaopt Net [57]	ResNet-12	41.10 ± 0.60	55.50 ± 0.60
DC [58]	ResNet-12	42.04 ± 0.17	57.63 ± 0.23
MDM-Net (Ours)	ResNet-12	**43.62 ± 0.46**	57.41 ± 0.33

and the training set of the FC100 data set belong to different categories, so their characteristics are more different in between. Therefore it is difficult to obtain a reliable prototype using conventional methods. However the data can be fully utilized by using multi-distance metrics method. The experimental results show that the multi-distance metric method and the adaptive distance loss function which we proposed are effective in the field of few shot learning.

5 Conclusion

A multi-distance metric net (MDM-Net) is proposed by considering multi-level features of the original image and the flipped image. Compared with the previous methods, the MDM-Net amplifies the subtle difference between samples and is more suitable to deal with few-shot learning task. The contributions of this paper can be summarized as following. Firstly, this paper proposes computing the distance between the images in multiple embedding spaces to obtain more reliable feature differences. Secondly, the multi-level feature maps are used to distinguish between different categories that are similar in features, because the shallow feature maps are utilized to help us pay more attention to the local details of the images. Thirdly, a weighted prototype is employed to replace the mean prototype, and is designed to weaken the influence of remote samples. Finally, a task-adaptive distance loss function is proposed which is beneficial to improve model performance. We tested different margins and selected a suitable value for 1-shot and 5-shot task respectively. In the future work, a deeper network could be employed to improve the feature extraction.

capability, and the feature maps should also be paid more attention. In addition, obtaining a more reasonable prototype also provides an effective way to improve the model performance.

Acknowledgements This work was supported by the National Key Research and Development Program of China (2018AAA0101601), and the Zhejiang Provincial Natural Science Foundation of China (LY20E050011).

References

1. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444
2. Krizhevsky A, Sutskever I, Hinton G. (2012) Imagenet classification with deep convolutional neural networks. In: Proceedings of advances in neural information processing systems, pp. 1097–1105
3. Deng J, Dong W, Socher R, et al. (2009) Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255
4. Szegedy C, Liu W, Jia Y, et al. (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9
5. Liu W, Anguelov D, Erhan D, et al. (2016) SSD: Single shot multibox detector. In: European conference on computer vision, pp. 21–37
6. Ren S, He K, Girshick R et al (2015) Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149
7. Redmon J, Divvala S, Girshick R, et al. (2016) You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779–788
8. He K, Gkioxari G, Dollár P, et al. (2017) Mask R-CNN. In: Proceedings of the IEEE international conference on computer vision, pp. 2961–2969
9. Kong T, Yao A, Chen Y, et al. (2016) Hypernet: towards accurate region proposal generation and joint object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 845–853

10. Sang J, Guo P, Xiang Z et al (2017) Vehicle detection based on faster-RCNN. J Chongqing Univ 40(7):32–36
11. Sharif Razavian A, Azizpour H, Sullivan J, et al. (2014) CNN features off-the-shelf: an astounding baseline for recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 806–813
12. Hawkins DM (2004) The problem of overfitting. J Chem Inf Comput Sci 44(1):1–12
13. Srivastava N, Hinton G, Krizhevsky A et al (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(1):1929–1958
14. Caruana R, Lawrence S, Giles CL. (2001) Overfitting in neural nets: backpropagation, conjugate gradient, and early stopping. In: Advances in neural information processing systems, pp. 402–408
15. Chawla NV, Bowyer KW, Hall LO et al (2002) SMOTE: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357
16. Salamon J, Bello JP (2017) Deep convolutional neural networks and data augmentation for environmental sound classification. IEEE Signal Process Lett 24(3):279–283
17. Cubuk ED, Zoph B, Mane D, et al. (2019) Autoaugment: learning augmentation strategies from data. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 113–123
18. Socher R, Ganjoo M, Manning CD, et al. (2013) Zero-shot learning through cross-modal transfer. In: Advances in neural information processing systems, pp. 935–943
19. Ma Y, Cambria E, Gao S (2016) Label embedding for zero-shot fine-grained named entity typing. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 171–180
20. Jayaraman D, Grauman K (2014) Zero-shot recognition with unreliable attributes. In: Advances in neural information processing systems, pp. 3464–3472
21. Pan SJ, Yang Q (2009) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359
22. Zamir AR, Sax A, Shen W, et al. (2018) Taskonomy: Disentangling task transfer learning. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp. 3712–3722
23. Long M, Zhu H, Wang J, et al. (2016) Unsupervised domain adaptation with residual transfer networks. In: Advances in neural information processing systems, pp. 136–144
24. Long M, Cao Y, Wang J, et al. (2015) Learning transferable features with deep adaptation networks. In: International conference on machine learning, pp. 97–105
25. Tzeng E, Hoffman J, Saenko K, et al. (2017) Adversarial discriminative domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7167–7176
26. Lake BM, Salakhutdinov RR, Tenenbaum J (2013) One-shot learning by inverting a compositional causal process. In: Advances in neural information processing systems, pp. 2526–2534
27. Xie R, Liu Z, Jia J, et al. (2016) Representation learning of knowledge graphs with entity descriptions. In: Thirtieth AAAI conference on artificial intelligence, pp. 2659–2665
28. Triantafillou E, Zemel R, Urtasun R. (2017) Few-shot learning through an information retrieval lens. In: Proceedings of the 31st international conference on neural information processing systems, pp. 2252–2262
29. Santoro A, Bartunov S, Botvinick M, et al. (2016) Meta-learning with memory-augmented neural networks. In: International conference on machine learning, pp. 1842–1850
30. Ravi S, Larochelle H. (2017) Optimization as a model for few-shot learning. In: Proceedings of the 5th international conference on learning representations. Toulon, France: ICLR, pp. 4077–4087
31. Oreshkin B, López PR, Lacoste A. (2018) Tadam: Task dependent adaptive metric for improved few-shot learning. In: Advances in neural information processing systems, pp. 721–731
32. Cheng G, Zhou P, Han J (2017) Duplex metric learning for image set classification. IEEE Trans Image Process 27(1):281–292
33. Zhang H, Berg AC, Maire M, et al. (2006) SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In: 2006 IEEE computer society conference on computer vision and pattern recognition, pp. 2126–2136
34. Liu X, Zhou F, Liu J et al (2020) Meta-Learning based prototype-relation network for few-shot classification. Neurocomputing 383:224–234
35. Vinyals O, Blundell C, Lillicrap T, et al. (2016) Matching networks for one shot learning. In: Proceedings of advances in neural information processing systems, pp. 3630–3638
36. Schroff F, Kalenichenko D, Philbin J (2015) FaceNet: a unified embedding for face recognition and clustering. In: Computer vision and pattern recognition, pp. 815–823
37. Oreshkin BN, Lopez PR, Lacoste A. (2018) TADAM: task dependent adaptive metric for improved few-shot learning. In: Neural information processing systems, pp. 721–731
38. Koch G, Zemel R, Salakhutdinov R (2015) Siamese neural networks for one-shot image recognition. In: ICML deep learning workshop, pp. 1–2
39. Snell J, Swersky K, Zemel R (2017) Prototypical networks for few-shot learning. In: Proceedings of advances in neural information processing systems. pp. 4077–4087
40. Li W, Wang L, Xu J, et al. (2019) Revisiting local descriptor based image-to-class measure for few-shot learning. In: Computer vision and pattern recognition, pp. 7260–7268
41. Li W, Wang L, Huo J, et al. (2020) Asymmetric distribution measure for few-shot learning. In: Proceedings of the twenty-ninth international joint conference on artificial intelligence, pp. 2957–2963
42. Chen H, Li H, Li Y, et al. (2021) Multi-level Metric Learning for Few-shot Image Recognition. arXiv:2103.11383; https://arxiv.org/abs/2103.11383v11383
43. Wang Y, Chao W, Weinberger KQ, et al. (2019) SimpleShot: Revisiting nearest-neighbor classification for few-shot learning. arXiv:1911.04623. https://arxiv.org/abs/1911.04623
44. Berthelot D, Carlini N, Goodfellow I, et al. (2019) MixMatch: a holistic approach to semi-supervised learning. In: neural information processing systems, pp. 5049–5059
45. Krizhevsky A, Nair V, Hinton G (2009) Cifar-10 and cifar-100 datasets. URl: https://www.cs.toronto.edu/kriz/cifar.html:
46. Munkhdalai T, Yu H (2017) Meta networks. In: Proceedings of the 34th international conference on machine learning, pp. 2554–2563
47. Finn C, Abbeel P, Levine S. (2017) Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the 34th international conference on machine learning, pp. 1126–1135
48. Finn C, Xu K, Levine S. (2018) Probabilistic model-agnostic meta-learning. In: Neural information processing systems, pp. 9516–9527
49. Satorras VG, Estrach JB (2018) Few-shot learning with graph neural networks. In: 6th international conference on learning representations, pp. 1–13
50. Sung F, Yang Y, Zhang L, et al. (2018) Learning to compare: relation network for few-shot learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1199–1208
51. Lee Y, Choi S. (2018) Gradient-based meta-learning with learned layerwise metric and subspace. In: International conference on machine learning, pp. 2927–2936
52. Mishra N, Rohaninejad M, Chen X, et al. (2018) A simple neural attentive meta-learner. In: 6th international conference on learning representations, pp. 1–17
53. Munkhdalai T, Yuan X, Mehri S, et al. (2018) Rapid adaptation with conditionally shifted neurons. In: International conference on machine learning, pp. 3661–3670

54. Liu Y, Lee J, Park M, et al. (2019) Learning to propagate labels: Transductive propagation network for few-shot learning. In: 7th international conference on learning representations, pp. 1–11
55. Sun Q, Liu Y, Chua T, et al. (2019) Meta-transfer learning for few-shot learning. In: Computer vision and pattern recognition, pp. 403–412
56. Chen W, Liu Y, Kira Z, et al. (2019) A closer look at few-shot classification. In: 7th international conference on learning representations., pp. 1–17
57. Lee K, Maji S, Ravichandran A, et al. (2019) Meta-learning with differentiable convex optimization. In: Computer vision and pattern recognition, pp. 10657–10665
58. Lifchitz Y, Avrithis Y, Picard S, et al. (2019) Dense classification and implanting for few-shot learning. In: Computer vision and pattern recognition, pp. 9258–9267

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.