Applied Intelligence (2022) 52:6148–6163
https://doi.org/10.1007/s10489-021-02769-6

Few-shot contrastive learning for image classification and its application to insulator identification

Liang Li[1] · **Weidong Jin**[1,2] · **Yingkun Huang**[1]

Accepted: 14 August 2021 / Published online: 2 September 2021

Abstract

This paper presents a novel discriminative Few-shot learning architecture based on batch compact loss. Currently, Convolutional Neural Network (CNN) has achieved reasonably good performance in image recognition. Most existing CNN methods facilitate classifiers to learn discriminating patterns to identify existing categories trained with large samples. However, learning to recognize novel categories from a few examples is a challenging task. To address this, we propose the Residual Compact Network to train a deep neural network to learn hierarchical nonlinear transformations to project image pairs into the same latent feature space, under which the distance of each positive pair is reduced. To better use the commonality of class-level features for category recognition, we develop a batch compact loss to form robust feature representations relevant to a category. The proposed methods are evaluated on several datasets. Experimental evaluations show that our proposed method achieves acceptable results in Few-shot learning.

Keywords Few-shot contrastive learning · Partial residual embedding module · Batch compact loss · Insulator identification

1 Introduction

Deep learning has remarkable performance in feature extraction and promising results in automobile self-driving [1], medical image analysis [2], and semantic segmentation [3]. In the wake of superior performance in feature extraction and representation, deep learning has also been introduced into image recognition [4, 5]. Mussina et al. [6] consider integrating multimodal information fusion for classifying possible situations by a fully connected network. Jiang et al. [7] propose multi-level perception to capture global and local representative features. Then, bounding box voting is utilized to generate the predictions. Liu et al. [8] segment images by Mask R-CNN network. Most image recognition methods heavily rely on a large volume of annotated datasets. Standard deep learning cannot identify the categories that do not appear in the train set. The model needs to be trained from scratch again to recognize a novel category. Besides, collecting enough labeled samples is generally time-consuming and laborious in deep learning.

Inspired by human beings′ remarkable ability to recognize novel objects after seeing only a handful of examples, Few-Shot Learning is proposed to tackle these problems. FSL can rapidly generalize to new tasks containing only a few samples with supervised information using prior knowledge. The core motivation of FSL is to learn to classify the instances from the training dataset correctly. Then the learned ability is applied to distinguish the novel instances of the test dataset. The data augmentation methods are used to maintain the consistency prediction of the instances [9, 10]. Through self-augmentation, knowledge forces each branch not to be over-confident in its predictions and improves the generalization ability. Owing to the knowledge learned from the source domain would be transferred to the target domain, the few-shot transfer learning framework is applied for machine fault diagnosis [11] and text classification [12].

Here we focus on the case of few-shot classification, where the given classification problem is assumed to contain only a handful of labeled examples per class. Few-shot classification usually involves a train set with base classes and a test set of novel classes. During training, K labeled samples for each of C unique classes from the train dataset

✉ Liang Li
liangli@my.swjtu.edu.cn

[1] Southwest Jiaotong University, Chengdu City, Sichuan Province, China

[2] China-ASEAN International Joint Laboratory of Integrated Transportation, Nanning University, Nanning City, Guangxi Province, China

being loaded into the model in one batch is defined as C-way K-shot FSL problem. For the C-way K-shot setting, any query feature needs to be compared with several representative features during training. Hence, it is essential to learn effectively from a small number of samples.

The hypothesis that the training data must be independent and identically distributed with the test data motivates us to use the correspondence between salient features and category information. As one of the main Overhead Catenary System (OCS) suspension structures, insulators realize electrical isolation and take on the mechanical loads. Therefore, enough attention should be paid to the maintenance of insulators. Because the working conditions of insulators are complex and changeable, the insulator states do not follow a specific expected pattern. The identification of insulator states improves the efficiency of prognostics and health management. Analyzing the cause of different states and potential tendency leads cleaning the pollution flashover regularly and setting the replacement plan for the low-value insulators intelligently. Since the Few-Shot Learning method can make full use of the same or different class sample pairs, it can also recognize the few test samples from the classes by comparing the feature similarities between different categories instead of just directly mapping features to a specific category.

The model should extract both the salient generalized features and the particular subtle features to enhance the generalization on the few-shot classification. Compared to the previous settings [13, 14], we further extend the query features to compare with both the sample features and other query features during training. With more features from the different categories, the model learns to construct better the discriminant subspace based on the class-level feature similarities. Compared with the class-level feature similarities, the query sample category is determined by the class label of the highest similarity. Secondly, we propose a novel deep learning architecture to extract representative features. Although residual learning addresses such an issue by introducing shortcut connections and identity mapping, shallow layers make sense in the training period instead of carrying gradients to all layers. Therefore, we preserve the shallow layer features which have high resolution to represent fine details of objects. The simple neural network is usually used in FSL to maintain the generalization. Previously most methods [13, 14]utilize four convolutional blocks for embedding modules. We extend to map the shallow layer features to the deep layers by the residual structure. The residual architecture is introduced to avoid degradation and consistently enhance the feature expression by extracting more practical features.

In this work, a novel Few-shot contrastive learning is proposed, and we apply the framework to identify insulator states. The main contributions are listed as follows:

1. To avoid degradation and consistently enhance the feature expression, we introduce the residual structure to map the shallow layer features to deep layers for refined image-level feature representation. By extracting more discriminative features, the framework better recognizes image-level images.
2. Unlike existing approaches [15, 16], we further consider more samples within a batch. Under the constraint of the loss function, we achieve the class-level feature representations. More samples contribute to constructing a more discriminative feature space where the relations are used to identify whether the selected samples are from the same category in every batch.
3. Extensive experiments on several datasets show the effectiveness of our proposed model for Few-Shot Learning.

The rest of the paper are described as follows. In Section 2, we review existing insulator identification, Few-Shot Learning, contrastive learning and metric learning. In Section 3, we describe our main algorithm and its implementation details. In Section 4, the experimental results are discussed and analyzed. In Section 5, we conclude the paper.

2 Related works

2.1 Insulator identification

As one of the main OCS suspension structures, insulators realize electrical isolation and take on the mechanical loads. Therefore, periodic inspection should be paid to the maintenance of insulators. Recently, deep learning has been introduced into insulator detection. The existing insulator detection methods have been successfully applied to the detection of insulator pollution degrees [4], insulator hydrophobicity degrees [17], and insulator icing degrees [18]. The ordinary methods collect samples of various categories and divide them into the train set, validation set, and the test set. The trained model can distinguish the existing categories well and conduct multiple classification of the samples to be tested in the test set. With more discriminative features and practical strategies, the insulator detection methods recognize insulators precisely.

However, the standard methods [19–21] require hundreds of samples per class to identify the existing categories. The limitation of samples makes it difficult to construct an accurate feature mapping network, leading to insufficient generalization. Since the Few-Shot Learning method can make full use of the same or different class sample pairs, it can also recognize the few test samples from the classes

by comparing the feature similarities between different categories instead of just directly mapping features to a specific category [22, 23].

2.2 Few-shot learning

Deep learning has remarkable performance in feature extraction and promising applications in automobile self-driving [1], medical image analysis [2], and fault diagnosis [3]. However, most deep learning algorithms heavily rely on a large volume of annotated sets. Collecting enough labeled samples is generally time-consuming and laborious in deep learning.

Inspired by human beings′ remarkable ability to recognize novel objects after seeing only a handful of examples, Few-Shot Learning (FSL) is proposed to tackle these problems. FSL is the complementation and expansion of deep learning. It mainly focuses on the feature representation and model structure efficient usage when the labeled data is limited. Unlike imbalanced classification, FSL trains and tests only with a few samples instead of all samples for a class. Previous outlier classification [24, 25] algorithms distinguish the similarities between samples and train samples, while FSL aims to distinguish samples between novel classes. FSL can quickly generalize to new tasks containing only a few samples with supervision class labels or semantic information using acquired knowledge or learning ability.

Currently, FSL can be categorized into three groups: transfer learning-based, optimization-based, and meta learning-based. Based on the assumption that the source domain will help the similar target domain, the transfer learning-based methods [26] take a pre-trained related model as a good initialization and adapt to a new task with a few iterations. Transferring well-trained source hypotheses in terms of parameters to learn the target hypothesis may be harmful to the target hypothesis. Larger weights are assigned to higher class-wise relevance instances [27] to alleviate negative transfer. Alternatively, only instances contributing to the target hypothesis are learned to revise the source hypothesis [28]. Both approaches focus on effective data selection. Such transfer learning methods have been applied in rotating machinery intelligent diagnosis [11] and new plant leaf and disease classification [29]. Most of them are only developed for a specific model. Thus, these methods lack the adaptation to various application scenes. Transductive learning [30] enhances the model generalization by unifying the representation of several classical intelligent models. The optimization-based methods [21] learn another neural network classifier that directly captures an optimization algorithm′s ability to have good generalization performance given only a set number of updates. Chelsea et al. [32] train good initial parameters for meta-learner to have good generalization on a new classification task. Based on [32], low-dimensional class-specific latent embeddings are decoded to generate the actual initial parameters [33]. The meta-learning methods learn a good metric or network in which the measurement of the target sets can be optimized continuously. Most of the researches focus on the similar measure of the samples in the metric space. By calculating the similarities between the samples [15, 34] or the similarities between the samples and class prototypes [16], the query images from new classes can be classified without further updating the network. The similarities between features can also be calculated by the feature direction in the feature space [35]. Then, it learns to combine these directions to obtain the principal direction for each novel class.

The designed neural network learns to distinguish between image pairs through similarity inference by constructing a meta-learning framework. Under the constraints of FSL, the number of input samples for each class is smaller than standard deep learning requirement. In each training iteration, we randomly select C classes from the train set with K labeled samples from each of the C classes to act as the sample set. The rest of those C classes′ samples act as the query set. For the 5-way 5-shot setting [13], the number of images from the sample set is 25 and the number of images from the query set is 50. All the query images need to calculate the similarities among the sample images. The similarity metric space will be further adjusted based on the similarities and category information.

The current FSL setting ignores the relationships within the sample set itself and the relationships within the query set itself. We propose to fully utilize the input samples and the additional structures during training.

2.3 contrastive learning

Contrastive learning focuses on the representation of features by comparing between different samples. The contrastive learning is designed with the original contrastive pair loss for discriminative architecture to gain enough consistent features to recognize and verify specified issues.

The contrastive pair loss aims at learning a discriminative feature space to measure the feature similarities among the input samples. Recently, contrastive learning has been introduced into predictions on unseen COVID-19 CT images [36, 37], scene text detection [38] and facial expression recognition [39]. Unlike simply mapping from features to a category, contrastive learning increases inter-class dispersion and intra-class compactness under different granularities. Through similarity learning constraints, the model can learn effective class-specific information for guiding more robust feature learning. It makes sure that samples from the same class have similar feature representations. On the contrary, feature representations

of different classes are far away in feature space. Most of the works focus on the positive pairs [37, 39, 40]. Few works focus on the negative pairs [41]. Indeed, the contrastive hard negative samples mining strategy enforces features embedded in a more discriminative feature space. The intra-class distance between the hard negative and the easy negative samples should be closer, and the inter-class distance between the hard negative and the positive samples should be increased during model updating. Most of the works classify the samples under the constraints of contrastive learning. Particularly, the original image, its augmented image and the rest of the images in the batch act as an anchor, a positive sample, and negative samples, respectively.

However, the contrastive learning methods mentioned above cannot be applied to FSL directly. Unlike standard contrastive learning, we take other samples of the same category in the same batch as positive samples and samples of other categories in the same batch as negative samples.

2.4 metric learning

Metric learning aims to learn a similarity metric that calculates the similarities of samples. A similarity metric is used to map similar samples closer and diverse samples far from each other. We can compare samples based on the underlying difference or the similarities of the images. The essence of metric learning is to obtain a transformation that can reflect the structural information of sample space or semantic constraint information so that the feature space and semantic space remain consistent. Therefore, metric learning has outstanding performance in judging the distance between data and classifying data. The content for metric learning can either be the distance [42] or the similarities [43, 44] between samples.

For image classification, metric learning optimizes the metric to make the distance of the sample pairs from the same category smaller than the distance between samples from different categories by using label information or image pair relation constraints. Metric learning methods generally use a linear projection, which subjects to solving real-world nonlinear characteristic problems. In recent years, deep metric learning, which provides a better solution for nonlinear data through activation functions, has attracted researchers′ attention in many areas.

To learning from raw data, deep metric learning [45] develops problem-based solutions through nonlinear subspace. The resigned trade-off factor is proposed to address the class-imbalance problem [46]. The positive pairs of small distances and negative pairs of large distances are simultaneously removed to improve learning efficiency and prediction accuracy. Generated semantically similar data [47] based on Linear Discriminant Analysis strengthen the feature representations in the metric space. For selecting different models, a discriminative stacked autoencoder [48] is applied to new features. Afterward, the model is fine-tuned by optimizing a new objective function. The nearest-neighbor search model [49] is proposed for searching different optimal nearest-neighbor numbers for different training instances. The success of these studies indicates the advantages of working in metric space.

The discriminative metric space is better constrained by increasing the size of data [50, 51]. Inspired by metric learning, the increased number of positives and negatives contributes to more discriminative boundaries. Thus, we use metric learning to construct the class-level feature space.

3 Methodology

3.1 FSL classification model

The meta-learning model learns to measure the similarities among different image pairs in the training period. The extracted representative features for one class should be distinct from other classes′ features. Since additional structural information can improve the model performance, the compact group takes group similarity instead of pair or triplet similarity. Therefore, the proposed FSL framework for image classification is named Residual Compact Network (RCNet). The input is all samples to be compared, and the outputs are the similarity scores of the given sample pairs.

The whole structure contains two main components: 1) the partial residual embedding module and 2) the relation module. The partial residual embedding module extracts distinctive features for each class under the batch compact loss constraint. The relation module measures the similarities among samples. With the help of the first module, each sample in the query set needs to be compared with samples in the sample set. Then, a similarity score represents the pair similarity. In this way, the representations and loss function complement the work with each other. Figure 1 represents the structure of the proposed Residual Compact Network.

The RCNet consists of supervised information and batch compact loss for few-shot image classification. According to the previous step, the RCNet aims to extract distinctive features of different classes and achieve similarity scores among given samples. The RCNet is a neural network designed for FSL. Therefore, its training method is different from those employed in standard deep learning. Instead of dividing the dataset into the train set, validation set, and test set, the dataset for FSL comprises the train set, support set, and query set. The train set has its own label space, and this space has no common intersection with the other two label

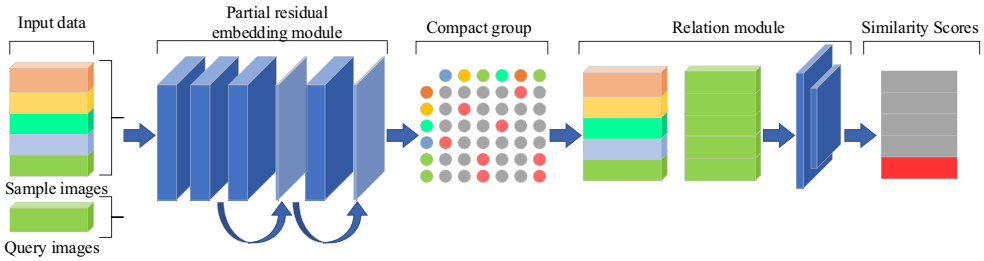

Fig. 1 Residual Compact Network

spaces. The support set and query set share the same label space. Since the query categories have no same category as the train categories, the performance of the trained classifier cannot be satisfactory. An effective way [13] is proposed to realize training via episode training. The strategy is choosing C classes with total KC samples randomly as a sample set and the rest of the samples in C classes as query set from the train set. Therefore, the corresponding mapping relationships learned from the train set can further fit in the query set. The partial residual embedding module $f(x)$ takes in sample x_i and x_j. Then, it produces feature maps $f(x_i)$ and $f(x_j)$, respectively. After concatenating the feature maps, the relation module g will produce the similarity score $r_{i,j}$ from 0 to 1 between sample x_i and x_j sequentially. For K shot setting, the class-level feature map is calculated with the element-wise sum of all samples from each class. We suppose that a similarity score should be higher if images are from the same class. Therefore, we use mean square error, shown in (1), to evaluate the relationships between similarity scores and class labels: images from same class should have a score of 1 and other conditions should have 0.

$$argmin \sum_{i=1} \sum_{j=1} (r_{i,j} - \mathbb{1}_{y_i=y_j})^2 \tag{1}$$

3.2 Partial residual embedding module

Both the completeness and uniqueness of the features are contributed by the feature extraction module in which the module must extract enough features belonging to a specified class. Computer vision has long been understood to follow a hierarchical process from the analysis of simple to complex features. Shallow layers in the neural network are sensitive to basic visual features while deeper layers capture basic shapes. With increasing depth [49], the network performs better at learning discriminative features and generalization on train data. However, the model performs worse as the network gets wider at a specific depth. Thus, it is necessary to choose the appropriate architecture for the different applications. The standard neural networks process inputs from low-level features up to task-specific high-level features. However, simply stacking more layers in the architecture may result in gradient vanishing or gradient exploding. Although these phenomena can be addressed by normalized initialization and nonlinear activation, adding more layers to an appropriate architecture will lead to the degradation of model performance. He et al. [52] propose residual learning to address such an issue by introducing shortcut connections and identity mapping. By splitting the residual network apart into several paths, although the residual network improves the model performance, it is shallow layers [53] making sense in the training period instead of carrying gradients to all layers. Therefore, the preservation of shallow layer representations may be vital in deep learning.

Inspired by [52, 53], we design the partial residual embedding module as a feature extraction architecture. In this module, we preserve the shallow layer representations and design shortcut connections in deep layers. The architecture is shown in Fig. 2, and more detailed specifications of the partial residual embedding module can be found in Table 1.

In the partial residual embedding module, every ConvBlock comprises a convolutional layer and a batch normalization layer. Due to the hierarchical structure, we preserve shallow layer feature maps which contain relative high-resolution natural information to describe the object better. Considering the character [54], we do not add ReLU activate function to all layers in the partial residual embedding module except the first layer. The input images are forwarded through the convolutional layer and the batch normalization layer sequentially. The ShortBlock is used as a shortcut connection that takes the feature maps from output of the

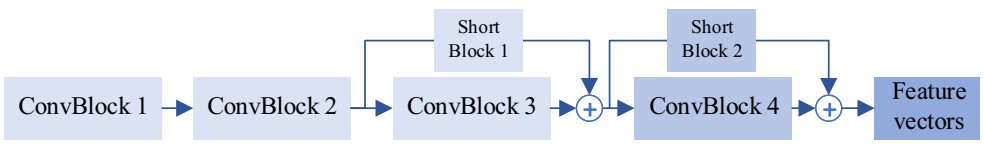

Fig. 2 Partial residual embedding module

Table 1 Specifications of the partial residual embedding module

Layers	Layer name	type	Depth	Stride	Padding
1	ConvBlock1	3 × 3Conv+BN+ReLU	64	1	1
2	ConvBlock2	3 × 3Conv+BN	64	1	1
		3 × 3Conv+BN	64	1	1
3	ConvBlock3	3 × 3Conv+BN	128	2	1
		3 × 3Conv+BN	128	1	1
	ShortBlock1	1 × 1Conv+BN	128	2	0
4	ConvBlock4	3 × 3Conv+BN	256	2	1
		3 × 3Conv+BN	256	1	1
	ShortBlock2	1 × 1Conv+BN	256	2	0
5	Feature Vectors	Pooling			

last layer and adds the current ConvBlock output as the final output. The feature vectors contain two parts: $m \times m$ feature map and its corresponding one-dimension representative vector. The $m \times m$ feature map is achieved after the ConvBlock4. The $m \times m$ feature map is adaptive pooling into a one-dimension vector to compare similarities among samples.

3.3 supervised contrastive learning

Our investigation is based on supervised contrastive learning [55], and we present this loss clearly before our batch compact loss.

3.3.1 supervised contrastive loss

For any network, based on the features extracted by its encoder module $f(x)$, the supervised contrastive loss for a positive pair of images (x_i, x_j) is defined as:

$$L = \frac{-1}{Ns} \sum_{j=1}^{m} \mathbb{1}_{i \neq j} \cdot \mathbb{1}_{y_i = y_j} \cdot log \frac{exp(f_N(x_i) \cdot f_N(x_j)/\tau)}{\sum_{k=1}^{n} \mathbb{1}_{i \neq k} exp(f_N(x_i) \cdot f_N(x_k)/\tau)} \tag{2}$$

where $f_N(x_i)$ is the normalization of the extracted feature $f(x_i)$. $f_N(x_i) \cdot f_N(x_k)$ is the similarity between $f(x_i)$ and $f(x_j)$ measured by cosine similarity. $\mathbb{1}_{y_i = y_j}$ is an indicator function evaluation to 1 if image x_j has the same class label to image x_i. Ns is the number of the total calculated pairs within a batch. We assume that the similarity score of the two similar images should be high. Minimizing the loss in (2) will increase the similarities among the similar representations. The τ denotes the predefined temperature scale parameter.

3.3.2 batch compact loss

Integrating problem-specific information and class labels in contrastive learning improves the effectiveness of the resulting supervised learning process and impact performance gains in downstream tasks. As mentioned before, the designed neural network learns to map similar features closer. The supervised contrastive loss encourages class-level representations to be similar for similar images. We further extend image-level representations to representations with this motivation. We propose this strategy that encourages the partial residual embedding module to extract general class-level representations within a batch. Using the batch compact loss between similar images from the same class is highly effective in learning the rich representations. The batch compact loss is shown in Fig. 3.

For a given image, this loss incentivizes the representations from other images of the same class to be similar. The batch compact loss for a set of given batch images is defined

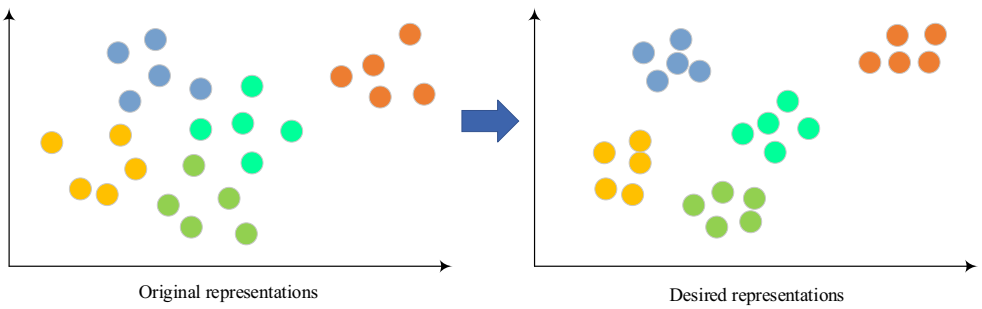

Fig. 3 Batch compact loss

as:

$$L^{cf} = \sum_{a=1}^{N} L_a^{cf} \tag{3}$$

$$L_a^{cf} = \frac{-1}{Nt} \sum_{j=1}^{m+n} \mathbb{1}_{i \neq j} \cdot \mathbb{1}_{y_i = y_j} \cdot log \frac{exp(f_N(x_i) \cdot f_N(x_j)/\tau)}{\sum_{k=1}^{m+n} \mathbb{1}_{i \neq k} exp(f_N(x_i) \cdot f_N(x_k)/\tau)} \tag{4}$$

In (4), L_a^{cf} is defined similarity between the image x_i and any image x_j from the same class in the ith batch. $f_N(x_i)$ is the normalization of the extracted feature $f(x_i)$. $f_N(x_i) \cdot f_N(x_j)$ is the similarity between $f(x_i)$ and $f(x_j)$ measured by cosine similarity. $\mathbb{1}_{y_i = y_j}$ is an indicator function evaluation to 1 if image x_j has the same class label to image x_i. τ denotes to a predefined temperature scale parameter. The batch compact loss considers all positive pairs both (x_i, x_j) and (x_j, x_i) in one batch and all the negative pairs in the same structures. $\mathbb{1}_{i \neq j}$ is an indicator function evaluation to 1 if compared image pair is not the same pair. Nt is the number of the total calculated pairs within a batch. The loss encourages the partial residual embedding module to give closely aligned representations to all entries from the same class in each instance of (4). For any anchor per batch in the 5-way 5-shot setting, the number of positives is 14 and the number of negatives is 60. In the standard supervised contrastive loss setting for FSL, the number of positives is 6 and the number of negatives is 8 for any anchor. The batch compact loss preserves the intention by adding more samples in both the positives and negatives under the FSL settings. With the increasing number of positives and negatives, the model can distinguish similar samples with better intra-class boundaries. The model trains with such a strategy can map similar features closer progressively.

3.3.3 metric loss functions

To evaluate the efficiency and validity of our proposed batch compact loss, we compare several commonly used loss functions in metric learning. Let x be the input feature vector, and y be its label. Let f be an encoder network mapping the input space to the embedding space and let $z = f(x)$ be the embedding vector.

$$L(z_a, z_p, z_n) = max(0, |z_a - z_p|^2 + |z_a - z_n|^2 + m) \tag{5}$$

The triplet loss [56] has been used to generate robust representations, which can only handle one positive and negative at a time. As shown in (5), z_a, z_p, z_n are the anchor vector, positive vector, and the negative vector, respectively. The parameter m is a margin parameter, and we set m to 1. The triplet loss pushes the negative sample outside of the boundary by the margin and keeps the positive sample within the boundary.

$$L(z_i, z_j) = log(1 + \sum_{k=1}^{2N} \mathbb{1}_{k \neq i} exp(z_i z_k - z_i z_j)) \tag{6}$$

The N-pair loss [57] is a generalization of triplet loss. It identifies a positive sample by comparing more than one negative sample, as shown in (6). z_i, z_j, z_k are the anchor vector, negative vectors, and positive vector, respectively. The N-pair loss pushes 2N-1 negative samples away simultaneously instead of one at a time.

To evaluate the efficiency of the proposed loss function, we experiment on several loss functions. Fashion-MNIST is commonly used to evaluate the efficiency of the loss function [58]. We conduct experiments to visualize the performance. Fashion-MNIST consists of 60000 train instances and 10000 test instances. Each example is a 28 × 28 grayscale image associated with a label from 10 classes. The result is shown in Fig. 4.

Based on the loss definition mentioned above, different loss functions extract meaningful features by different data structures. Both the n-pairs loss function and triplet loss function are based on a simple pair of samples. In contrast, the supervised contrastive loss and batch compact loss are based on complex pairs of samples. Both the supervised contrastive loss and batch compact loss construct the metric space through multiple sample samples. Unlike the first two loss functions, there are more positive samples to constrain the similarities in each iteration. We set the batch size for batch compact loss to 75 and other loss functions to 50. This setting is to stimulate data usage in the FSL. Model benefits from more structural features and larger batch size. The contrastive loss and the batch compact loss have better performance than the other loss functions. The batch compact loss performs better than the supervised contrastive loss at the first several epochs by a little margin. In the whole training period, we can observe a slight margin at the twentieth epoch. With more training epochs, more informative samples are extracted to construct the discriminative metric space.

Single objective loss is just a degenerate case of the multi-objective loss. A simple method is to form a composite objective function as the weighted sum of the objectives. The weight for an objective is proportional to the preference factor assigned to that objective. Without any knowledge of the potential trade-off solutions, this is an even more difficult task. Standard multi-objective optimization methods convert multiple objectives into a single objective using a weighted-sum method. The strategy scalarizes a set of objectives into a single objective by pre-multiplying each objective with a user-supplied weight. The procedure cannot be used to find Pareto-optimal solutions

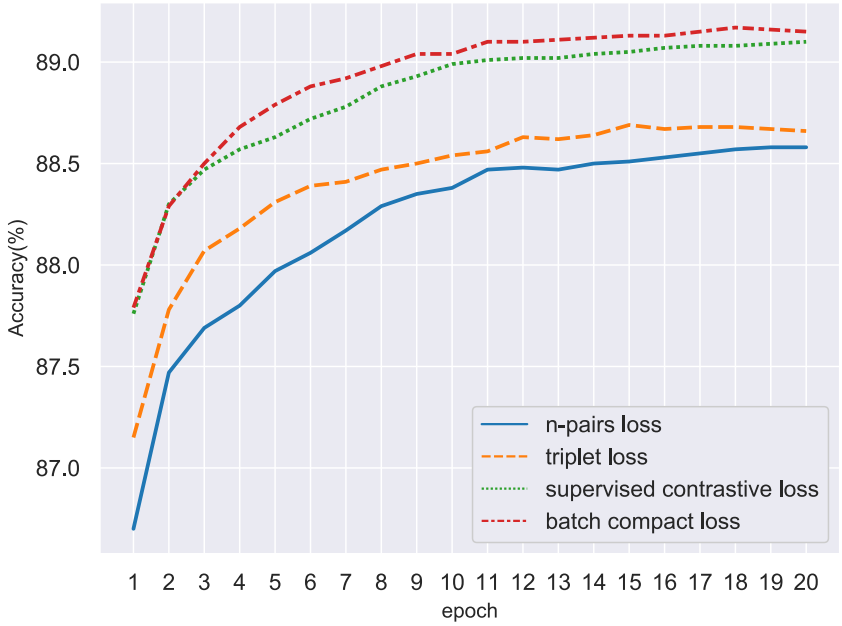

Fig. 4 Comparison of different loss functions

that lie on the non-convex portion of the Pareto-optimal front. Although there exist approaches addressing non-convex objective problems, we split RCNet training into two-stage training procedures. The details of the training algorithm are shown in Tables 2 and 3.

Based on different FSL settings, the partial residual embedding module is trained from scratch according to Table 2. In this stage, class-level representations will be more general iteratively. After the training is finished, the partial residual embedding module weights are saved for the next stage.

In the second stage, the relation module based on MSE loss is updated to identify the similarities among the samples iteratively. Unlike standard deep learning, the proposed model learns to extract class-level representations and measure the similarities between novel categories.

4 Experiments

To demonstrate the effectiveness of our method and inspect the effects of using different C-way K-shot variants, we extensively evaluate our proposed RCNet with three public FSL datasets and one OCS insulator sub-dataset. In the subsequent sections, we will present the datasets used, experimental settings, and results with discussions.

Table 2 The first stage training algorithm

Partial residual embedding module training algorithm

Input: few-shot setting C,K,Q, sample set Ds, Query set Dq, partial residual embedding module f, relation module g, constant τ

For episode $\in$ number of training iteration do

For randomly sampled image $\{x_i\}_{i=1}^{CK} \in$ Ds , $\{x_j\}_{j=1}^{CQ} \in$ Dq do

Sampled images set $X := \{x_i\}_{i=1}^{CK} \cup \{x_j\}_{j=1}^{CQ}$

Sampled labels set $Y := \{y_i\}_{i=1}^{CK} \cup \{y_j\}_{j=1}^{CQ}$

$\{hs\}, \{hq\} = f(X, X), H := \{hs\} \cup \{hq\}$

For all $i \in \{1, \ldots, CK + CQ\}$ and $j \in \{1, \ldots, CK + CQ\}$ do

$f_N(x_i) = h_i / \|h_i\|, f_N(x_j) = h_j / \|h_j\|, h \in H$

$L_a^{cf} = \frac{-1}{Nt} \sum_{j=1}^{m+n} \mathbb{1}_{i \neq j} \cdot \mathbb{1}_{y_i = y_j} \cdot log \frac{exp(f_N(x_i) \cdot f_N(x_j)/\tau)}{\sum_{k=1}^{m+n} \mathbb{1}_{i \neq k} exp(f_N(x_i) \cdot f_N(x_k)/\tau)}$

$L^{cf} = \sum_{a=1}^{N} L_a^{cf}$

Update f to minimize L^{cf}

End for

End for

Return updated partial residual embedding module f

Table 3 The second stage training algorithm

Relation module training algorithm

Input: few-shot setting C,K,Q, sample set Ds, Query set Dq, partial residual embedding module f, relation module g

For episode ∈ number of training iteration do

For randomly sampled image $\{x_i\}_{i=1}^{CK} \in$ Ds , $\{x_j\}_{j=1}^{CQ} \in$ Dq do

Sampled images set $X := \{x_i\}_{i=1}^{CK} \cup \{x_j\}_{j=1}^{CQ}$

Sampled labels set $Y := \{y_i\}_{i=1}^{CK} \cup \{y_j\}_{j=1}^{CQ}$

$\{hs\}, \{hq\} = f(X, X)$, $H := \{hs\} \cup \{hq\}$

For all $i \in \{1, \ldots, CK + CQ\}$ and $j \in \{1, \ldots, CK + CQ\}$ do

Similar score $r_{i,j} = g(\{hs\} \cup \{hq\})$

$L = \sum_{i=1} \sum_{j=1} (r_{i,j} - \mathbb{1}_{y_i = y_j})^2$

Update f,g to minimize L

End for

End for

End for

Return updated partial residual embedding module f and relation module g

4.1 Dataset

We tested our method on three datasets: three public FSL datasets and one OCS insulator sub-dataset.

4.1.1 Omniglot

The Omniglot dataset [59] is a handwritten character dataset that contains 50 classes and the total number is 1623 different characters drawn by 20 people. With data augmentation, the existing data are augmentated by 90, 180 and 270 degrees of rotation, the train set contains 1200 original classes and their corresponding augmentations. The remaining 423 classes and their corresponding augmentations constitute the test set. All images in Omniglot are resized to 28 × 28 pixels.

4.1.2 miniImagenet

The miniImagenet dataset is a subclass of the ILSVRC-12 dataset proposed by [13]. This dataset has 100 classes with each having 600 images. All images are 84 × 84 pixels. We follow [31] and split the dataset into 64, 16 and 20 classes for training, support and testing, respectively. These separated datasets have no intersection.

4.1.3 tieredImagenet

The tieredImagenet dataset [60] is a larger subset of the ILSVRC-12 dataset. The dataset is split into 20 train, 6 validation, and 8 test categories. Each category contains between 10 and 30 classes. All images are 224 × 224 pixels. This dataset groups classes into broader categories corresponding to higher-level nodes in the ImageNet hierarchy. This division ensures that all the train classes are sufficiently distinct from the test classes. Additionally, the tiered structure of the tieredImagenet dataset may be helpful for hierarchical relationships between classes.

4.1.4 OCS insulator

All the images in this dataset are obtained by the catenary checking video monitor system of the Beijing-Ganzhou high-speed rail line. The insulators are taken in different illumination, angle of cameras. To avoid creating an imbalanced dataset that may lead to ambiguous accuracies, we build a balanced dataset by data augmentation method [61]. We divide the insulators into five categories, as shown in Fig. 5. Figure 5a-e shows the normal, umbrella petticoats damaged and foreign body, polluted, and broken states of insulators, respectively. Motivated by [22], we randomly select 9 samples, 7 samples and 50 samples per class for training, validation and testing, respectively. All images are 224 x 224 pixels.

4.2 Implementation details

Our method is implemented by Ubuntu 16.04, CUDA 10.1 and Pytorch 1.4. The hardware of the experiment is Intel Xeon X5690, Nvidia GeForce GTX 1080ti and 32GB RAM. For the whole network architecture, we train the model from scratch without fine-tuning. Due to the limitation of the GPU memory, we resize images in tieredImagenet and OCS insulator sub-dataset to 84 × 84 pixels.

The first training stage aims at training the partial residual embedding module based on the batch compact

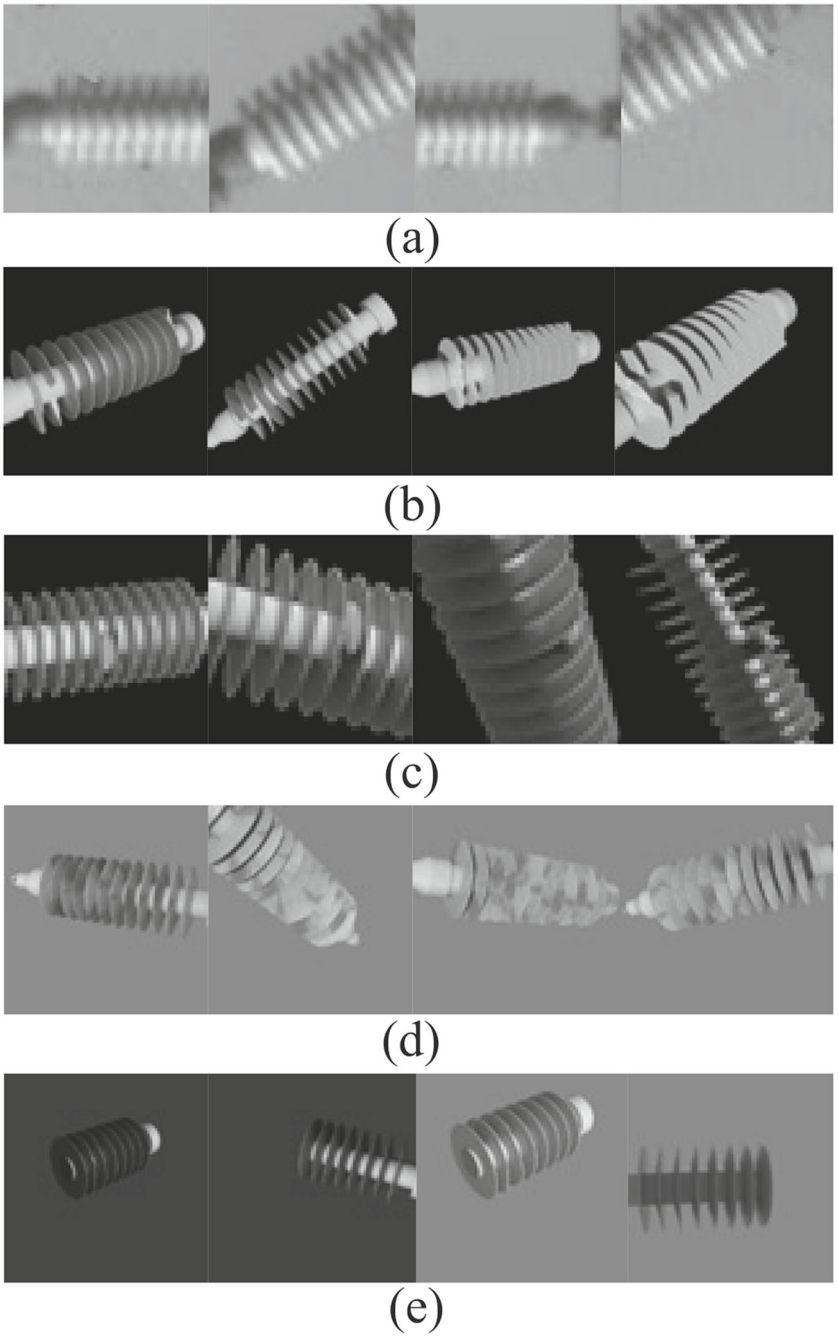

Fig. 5 States of different types of insulators. **a** Normal. **b** umbrella petticoats damaged. **c** foreign body. **d** polluted. **e** broken

loss. Due to the different C-way K-shot settings, the number of query set varies. During training, the learning rate is 0.5 with a decay rate 0.1 of every 20k epochs. The model is trained for 150k epochs. The optimization is Adam and this step is optimized by batch compact loss. We set τ to 0.9.

The second stage aims at training the relation module based on MSE loss. During training, the learning rate is 0.001 and the learning decay rate is 0.5 of every 100k epochs. The model is trained for 500k epochs. The optimization is Adam and this stage is optimized by MSE loss. Using the same dataset in the first stage, the pre-trained partial residual embedding module weight is loaded and the RCNet framework is trained based on MSE loss sequentially. After training in this stage, the RCNet predicts the label of test samples according to the similarity scores. The accuracy is calculated with 95% confidence intervals

over test episodes according to different settings. We set the test episode to 1000 for the Omniglot dataset and 600 for the other datasets.

4.3 Experimental results and analysis

4.3.1 Experiments on different structures

To select the appropriate network structure, we conduct experiments on different structures before the actual training. A group of comparative experiments is conducted to analyze the effects of depth and convolution kernels.

Figure 6 shows the accuracy of different partial residual modules. The abscissa represents the results of every five thousand iterations, and the ordinate represents the accuracy of the training process. We explore the performances of different partial residual modules on the miniImagenet dataset to select a generalized architecture. The specific parameters for each dataset will be adjusted according to the specific dataset. For all modules, the S7 module outperforms others in training from scratch by a sufficient margin. We apply the S7 module to our architecture, as shown in Table 4. The detailed specifications of the different partial residual modules can be found in Table 4. Layer 2 to layer 4 in Table 1 can be replaced with different partial residual module settings in Table 4. In FSL, the shallow module performs better than the deeper ones. In the same depth, more convolution kernels are more helpful for the network to extract useful features. The appropriate module depth and convolution kernels can effectively improve the model performance.

Table 4 Specifications of the partial residual embedding module

Name	Framework	Blocks
S1	Conv64|Conv64|Conv64|Conv64	4
S2	2×Conv64|2×Conv64|2×Conv64|2×Conv64	4
S3	Conv64|Conv128|Conv256|Conv512	4
S4	2×Conv64|2×Conv128|2×Conv256|2×Conv512	4
S5	Conv64|Conv64|Conv64	3
S6	2×Conv64|2×Conv64|2×Conv64	3
S7	Conv64|Conv128|Conv256	3
S8	2×Conv64|2×Conv128|2×Conv256	3

4.3.2 Experiments on different datasets

The first set of experiments is conducted on the Omniglot dataset. Because of the limited written characters, we evaluate our method both in 5-way and 20-way settings. These results are reported in Table 5. Symbol '-' indicates the unreported result. Our method achieves acceptable performance among several algorithms. Specifically, our method improves the 5-way 1-shot, the 20-way 5-shot accuracy by 0.1%, and 20-way 1-shot accuracy by 0.4%. However, comparing to the 5-way 5-shot settings, our method falls behind by 0.1% to the best previously published results. With batch compact loss, our method improves the performance demonstrate that the increased numbers of positives and negatives contribute to a distinctive feature space under the FSL setting.

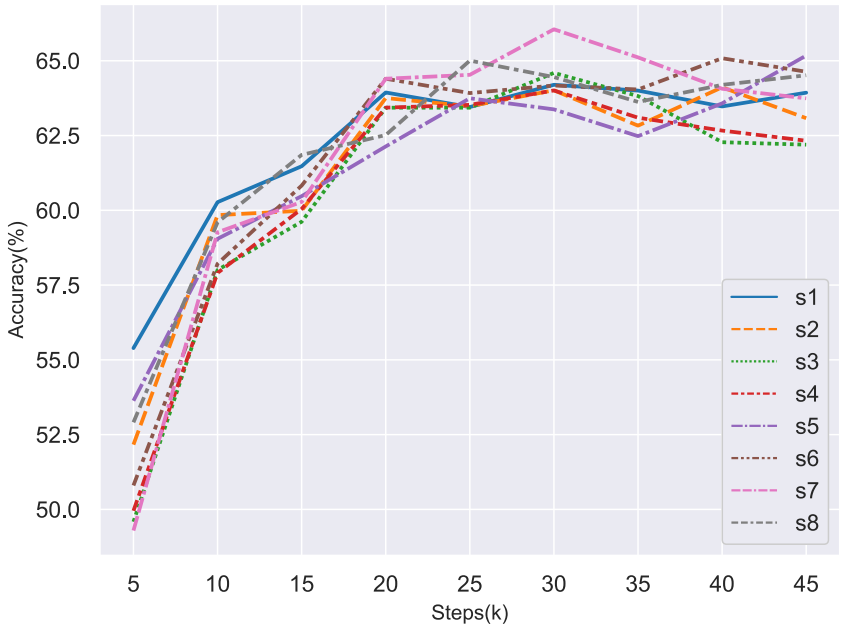

Fig. 6 Accuracy of different residual modules

Table 5 The accuracy on Omniglot dataset

Model	FT	5-way Acc		20-way Acc	
		1-shot	5-shot	1-shot	5-shot
MANN [62]	N	82.8%	94.9%	-	-
MATCHING NETS [13]	N	98.1%	98.9%	93.8%	98.5%
MATCHING NETS [13]	Y	97.9%	98.7%	93.5%	98.7%
CNAPS [63]	Y	97.4±0.3%	99.4±0.1%	95.3±0.2%	98.4±0.1%
SIAMESE NETS WITH MEMORY [64]	N	98.4%	99.6%	95.0%	98.6%
NEURAL STATISTICIAN [65]	N	98.1%	99.5%	93.2%	98.1%
META NETS [66]	N	99.0%	-	97.0%	-
PROTOTYPICAL NETS [16]	N	98.8%	99.7%	96.0%	98.9%
MAML [32]	Y	98.7±0.4%	**99.9±0.1%**	95.8±0.3%	98.9±0.2%
RELATION NET [15]	N	99.6±0.2%	99.8±0.1%	97.6±0.2%	99.1±0.1%
RCNet	N	99.6±0.2%	99.7±0.1%	97.7±0.2%	99.1±0.1%
RCNet with batch compact loss	N	**99.7±0.2%**	99.8±0.1%	**98.0±0.1%**	**99.2±0.1%**

Considering the unknown accuracy distribution, we use the Wilcoxon signed-rank test [67] to determine whether the accuracy series come from the same distribution. The null hypothesis is that the approaches to be compared are similar in performance, and the alternative hypothesis is that the approaches to be compared are not. A p-value of less than 0.05 indicates statistical significance between the compared approaches. In the first set of experiments, the bold results in the table are the best results with significant differences to other approaches (only the coded methods are tested).

The second set of experiments is conducted on the miniImagenet dataset. We evaluate our method both in the 5-way 1-shot and the 5-way 5-shot. These results are reported in Table 6. Symbol '-' indicates unreported result. Specifically, our model improves the 5-way 1-shot accuracy by 3.6% and the 5-way 5-shot accuracy by 0.7% to the best previously published results. On the one hand, under the batch compact loss constraint, the model learns to map the similar features closer and construct better the discriminant subspace based on the class-level feature similarities. On the other hand, the partial residual embedding module preserves the shallow layer features. With richer low-level features, the RCNet better describes the object-level samples. Our method, benefiting from the batch compact loss, further improves the ability to recognize the novel categories based on the similarities. Under the FSL settings, the class with fewer samples in each category benefits more from the batch compact loss. In this experiment, the p-value for all comparisons is 0.002. Because the p-value is less than 0.05, the null hypothesis is rejected. It means that our model is better than other approaches (only the coded methods are tested).

The third set of experiments is conducted on the tieredImagenet dataset. We evaluate our method both in the 5-way 1-shot and the 5-way 5-shot. These results are reported in Table 7. Since the train classes are sufficiently

Table 6 The accuracy on miniImagenet dataset

Model	FT	5-way Acc	
		1-shot	5-shot
MATCHING NETS [13]	N	43.56±0.84%	55.31±0.73%
META-LEARN LSTM [31]	N	43.44±0.77%	60.60±0.71%
MAML [32]	Y	48.70±1.84%	63.11±0.92%
META NETS [66]	N	49.21±0.96%	-
PROTOTYPICAL NETS [16]	N	49.42±0.78%	68.20±0.66%
RELATION NET [15]	N	50.44±0.82%	65.32±0.70%
CovaMNet [68]	N	51.19±0.76%	67.65±0.63%
RCNet	N	51.33±0.86%	67.69±0.79%
RCNet with batch compact loss	N	**54.85±0.84%**	**68.92±0.77%**

Table 7 The accuracy on tieredImagenet dataset

Model	FT	5-way Acc	
		1-shot	5-shot
MAML [32]	Y	51.67±1.81%	70.30±1.75%
SSL [60]	N	52.39±0.44%	69.88±0.22%
PROTOTYPICAL NETS [16]	N	53.31±0.89%	72.69±0.74%
CovaMNet [68]	N	54.07±0.91%	70.34±0.75%
RELATION NET [15]	N	54.48±0.93%	71.31±0.78%
RCNet	N	56.92±0.97%	73.41±0.80%
RCNet with batch compact loss	N	**58.42±0.96%**	**74.17±0.78%**

distinct from the test classes, the methods perform better on this dataset than the second experiment. Specifically, our method improves the 5-way 1-shot accuracy by nearly 4% and the 5-way 5-shot accuracy by nearly 1.5% to the best previously published results. The class-level features obtained by the proposed architecture can effectively construct the discriminative feature space. With the distinct class-level boundaries, our method better distinguishes the different novel categories. In this experiment, the p-value for all comparisons is 0.002. Because the p-value is less than 0.05, the null hypothesis is rejected. Therefore, we conclude that our model performs better than other approaches (only the coded methods are tested).

The fourth set of experiments is conducted on the OCS insulator sub-dataset. Considering the limited categories and samples in this dataset, we apply data augmentation including randomly crop all the images into 84 × 84 and color transformation to ease the overfitting. We evaluate these models both in the 5-way 1-shot and the 5-way 5-shot settings. Based on FSL, all the models identify the different insulator states even with only one or few samples per class. We only train the model for 30 epochs in the first stage to avoid falling into a local optimum. Then, we train our model for 30K epochs. All other models are trained for 30K epochs. These results are reported in Table 8. Specifically, our method improves the 5-way 1-shot accuracy by 0.52% and the 5-way 5-shot accuracy by 0.57% to the other methods applied in this dataset. The p-value for all comparisons is 0.002. Because the p-value is less than 0.05, the null hypothesis is rejected. It means that our approach outperforms other approaches not by accident.

Table 8 The accuracy on OCS insulator sub-dataset

Model	FT	5-way Acc	
		1-shot	5-shot
MAML [32]	N	81.44±1.24%	88.93±0.68%
MATCHING NETS [13]	N	82.12±0.62%	86.79±0.23%
META-LEARN LSTM [31]	N	84.16±0.19%	87.80±0.13%
CovaMNet [68]	N	84.93±0.91%	87.81±0.33%
PROTOTYPICAL NETS [16]	N	83.77±0.14%	86.69±0.11%
RELATION NET [15]	N	84.31±0.43%	89.11±0.28%
RCNet	N	85.21±0.65%	89.59±0.36%
RCNet with batch compact loss	N	**85.45±0.57%**	**89.68±0.33%**

5 Conclusion

In this paper, we focus on the challenge of the few-shot image classification problem via comparing discriminative class-level features from a few labeled examples. The partial residual embedding module and batch compact loss are our two contributions. The partial residual embedding module utilizes low-level features and shortcut connections to generate discriminative features. The shortcut connections preserve the shallow layer features and transfer them to the high layers. For few-shot image classification, we manage to learn a more robust class-level feature extractor through the training period. The batch compact loss exploits the features within one batch fully. Extensive experiments on several datasets show the effectiveness of our proposed model for FSL.

References

1. Anselme N, TN H, Hyeon KD, Tae KK, Seon HC (2021) Deep learning based caching for Self-Driving cars in Multi-Access edge computing. IEEE Trans Intell Transp Syst 22(5):2862–2877. https://doi.org/10.1109/tits.2020.2976572
2. Justin K, Lipo W, Jai R, Tchoyoson L (2018) Deep learning applications in medical image analysis. IEEE Access 6:9375–9389. https://doi.org/10.1109/access.2017.2788044
3. Alberto G-G, Sergio O-E, Sergiu O, Victor V-M, Pablo M-G, Jose G-R (2018) A survey on deep learning techniques for image

and video semantic segmentation. Appl Soft Comput 70:41–65. https://doi.org/10.1016/j.asoc.2018.05.018
4. Yan Q, Guangning W, Zhang X, Yujun G, Xueqin Z, Kai L (2019) An Extreme-Learning-Machine-Based hyperspectral detection method of insulator pollution degree. IEEE Access 7:121156–121164. https://doi.org/10.1109/access.2019.2937885
5. Zhenbing Z, Xiaoqing F, Guozhi X, Lei Z, Yincheng Q, Ke Z (2017) Aggregating deep convolutional feature maps for insulator detection in infrared images. IEEE Access 5:21831–21839. https://doi.org/10.1109/access.2017.2757030
6. Damira M, Aidana I, JP K, Mehdi B (2020) Multi-Modal Data fusion using deep neural network for condition monitoring of high voltage insulator. IEEE Access 8:184486–184496. https://doi.org/10.1109/access.2020.3027825
7. Hao J, Xiaojie Q, Jing C, Xinyu L, Xiren M, Shengbin Z (2019) Insulator fault detection in aerial images based on ensemble learning with Multi-Level perception. IEEE Access 7:61797–61810. https://doi.org/10.1109/access.2019.2915985
8. Wenqiang L, Zhigang L, Hui W, Zhiwei H (2020) An automated defect detection approach for catenary Rod-Insulator textured surfaces using unsupervised learning. IEEE Trans Instrum Meas 69(10):8411–8423. https://doi.org/10.1109/tim.2020.2987503
9. Shixin H, Xiangping Z, Si W, Zhiwen Y, Mohamed A, Hau-San W (2021) Behavior regularized prototypical networks for semi-supervised few-shot image classification. Pattern Recogn 112:107765–107775. https://doi.org/10.1016/j.patcog.2020.107765
10. Jin-Woo S, Hong-Gyu J, Seong-Whan L (2021) Self-augmentation: Generalizing deep networks to unseen classes for few-shot learning. Neural Netw 138:140–149. https://doi.org/10.1016/j.neunet.2021.02.007
11. Jingyao W, Zhibin Z, Chuang S, Ruqiang Y, Xuefeng C (2020) Few-shot transfer learning for intelligent fault diagnosis of machine. Measurement 166:108202–108214. https://doi.org/10.1016/j.measurement.2020.108202
12. Chongyu P, Jian H, Jianxing G, Xingsheng Y (2019) Few-Shot Transfer learning for text classification with lightweight word embedding based models. IEEE Access 7:53296–53304. https://doi.org/10.1109/access.2019.2911850
13. Oriol V, Charles B, Timothy L, Koray K, Daan W (2016) Matching networks for one shot learning. In: Advances in Neural Information Processing Systems, vol 29. Curran Associates, Barcelona, pp 3630–3638
14. Zhong J, Xingliang C, Yunlong Y, Yanwei P, Zhongfei Z (2020) Improved prototypical networks for few-Shot learning. Pattern Recogn Lett 140:81–87. https://doi.org/10.1016/j.patrec.2020.07.015
15. Flood S, Yongxin Y, Li Z, Tao X, Philip HST, Timothy MH Learning to Compare: Relation Network for Few-Shot Learning. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, Salt Lake City, pp 1199–1208. https://doi.org/10.1109/CVPR.2018.00131
16. Jake S, Kevin S, Richard SZ Prototypical Networks for Few-shot Learning. In: 31st International Conference on Neural Information Processing Systems. Curran Associates Inc, Long Beach, pp 4080–4090. https://doi.org/10.5555/3294996.3295163
17. KC-C A, Thanos T, KV T, Giorgos S, GI F (2020) Hydrophobicity classification of composite insulators based on convolutional neural networks. Eng Appl Artif Intell 91:103613–103622. https://doi.org/10.1016/j.engappai.2020.103613
18. Yanqing L, Lichun S, Qin H, Xingliang J, Meilin Z, Zhou Y, Hanxiang L (2021) Statistical analysis on the DC discharge path of ice-covered insulators under natural conditions. Int J Electr Power Energy Syst 130:106961–106967. https://doi.org/10.1016/j.ijepes.2021.106961
19. Sampedro C, Rodriguez-Vazquez J, Rodriguez-Ramos A, Carrio A, Campoy P (2019) Deep Learning-Based system for automatic recognition and diagnosis of electrical insulator strings. IEEE Access 7:101283–101308. https://doi.org/10.1109/access.2019.2931144
20. Xian T, Dapeng Z, Zihao W, Xilong L, Hongyan Z, De X (2020) Detection of power line insulator defects using aerial images analyzed with convolutional neural networks. IEEE Trans Syst Man Cybern-Syst 50(4):1486–1498. https://doi.org/10.1109/tsmc.2018.2871750
21. Diana S, Damira P, Mehdi B, Alex J (2020) IN-YOLO: Real-Time detection of outdoor high voltage insulators using UAV imaging. IEEE Trans Power Deliv 35(3):1599–1601. https://doi.org/10.1109/tpwrd.2019.2944741
22. Ansi Z, Shaobo L, Yuxin C, Wanli Y, Rongzhi D, Jianjun H (2019) Limited data rolling bearing fault diagnosis with Few-Shot learning. IEEE Access 7:110895–110904. https://doi.org/10.1109/access.2019.2934233
23. Sonal D, VN K, GA K (2021) Intelligent fault diagnosis of rotary machines: Conditional auxiliary classifier GAN coupled with meta learning using limited data. IEEE Trans Instrum Meas 70:1–11. https://doi.org/10.1109/tim.2021.3082264
24. Toshitaka H, Hamido F, Andres H-M (2021) Less complexity one-class classification approach using construction error of convolutional image transformation network. Inf Sci 560:217–234. https://doi.org/10.1016/j.ins.2021.01.069
25. Toshitaka H, Hamido F (2020) Cluster-based zero-shot learning for multivariate data. J Ambient Intell Human Comput 12(2):1897–1911. https://doi.org/10.1007/s12652-020-02268-5
26. Zhaohong D, Yizhang J, Hisao I, Kup-Sze C, Shitong W (2016) Enhanced Knowledge-Leverage-Based TSK fuzzy system modeling for inductive transfer learning. ACM Trans Intell Syst Technol 8(1):1–21. https://doi.org/10.1145/2903725
27. Siwei F, DM F (2019) Few-shot learning-based human activity recognition. Expert Syst Appl 138:112782–112793. https://doi.org/10.1016/j.eswa.2019.06.070
28. Wenhe L, Xiaojun C, Yan Y, Yi Y, HA G (2018) Few-Shot Text and image classification via analogical transfer learning. ACM Trans Intell Syst Technol 9(6):1–20. https://doi.org/10.1145/3230709
29. David A, Artzai P, Unai I, Alfonso M, s-EM, G, Arantza B, Aitor A-G (2020) Few-Shot Learning approach for plant disease classification using images taken in the field. Comput Electron Agric 175:105542–105549. https://doi.org/10.1016/j.compag.2020.105542
30. Lixiao X, Zhaohong D, Peng X, Kup-Sze C, Shitong W (2019) Generalized Hidden-Mapping transductive transfer learning for recognition of epileptic electroencephalogram signals. IEEE Trans Cybern 49(6):2200–2214. https://doi.org/10.1109/TCYB.2018.2821764
31. Sachin R, Hugo L (2017) Optimization as a Model for Few-Shot Learning. In: 5th International Conference on Learning Representations, Toulon. OpenReview.net
32. Chelsea F, Pieter A, Sergey L Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In: Proceedings of the 34th International Conference on Machine Learning. PMLR, Sydney, pp 1126–1135
33. RA A, Dushyant R, Jakub S, Oriol V, Razvan P, Simon O, Raia H (2019) Meta-learning with Latent Embedding Optimization. In: 7th International Conference on Learning Representations. OpenReview.net, New Orleans

34. Junbo L, Yaping H, Mei ZQT (2019) Learning visual similarity for inspecting defective railway fasteners. IEEE Sens J 19(16):6844–6857. https://doi.org/10.1109/jsen.2019.2911015
35. Shafin R, Salman K, Fatih P (2018) A Unified approach for Conventional Zero-shot, Generalized Zero-shot and Few-shot Learning. IEEE Trans Image Process 27(11):5652–5667. https://doi.org/10.1109/TIP.2018.2861573
36. Mohammad S, Hossain MS (2021) MetaCOVID: A Siamese neural network framework with contrastive loss for n-shot diagnosis of COVID-19 patients. Pattern Recogn 113:107700–107710. https://doi.org/10.1016/j.patcog.2020.107700
37. Xiaocong C, Lina Y, Tao Z, Jinming D, Yu Z (2021) Momentum contrastive learning for few-shot COVID-19 diagnosis from chest CT images. Pattern Recogn 113:107826–107833. https://doi.org/10.1016/j.patcog.2021.107826
38. Ran W, Yaoyi L, Haiyan L, Ze T, Hongtao L, Nengbin C, Xuejun Z (2021) A robust and effective text detector supervised by Contrastive Learning. IEEE Access:26431–26441. https://doi.org/10.1109/access.2021.3057108
39. Haifeng Z, Wen S, Zengfu W (2020) Weakly supervised Local-Global attention network for facial expression recognition. IEEE Access 8:37976–37987. https://doi.org/10.1109/access.2020.2975913
40. Junling G, Lei X, Ayache B, Mingxi W (2019) A deep Siamese-Based plantar fasciitis classification method using shear wave elastography. IEEE Access 7:130999–131007. https://doi.org/10.1109/access.2019.2940645
41. Siyuan Y, Hua Z, Wenqi R, Chao M, Xiaoguang H, Xiaochun C (2021) Robust online tracking via contrastive Spatio-Temporal aware network. IEEE Trans Image Process 30:1989–2002. https://doi.org/10.1109/TIP.2021.3050314
42. Bac N, Carlos M, Bernard DB (2018) Distance metric learning for ordinal classification based on triplet constraints. Knowl-Based Syst 142:17–28. https://doi.org/10.1016/j.knosys.2017.11.022
43. Jianqing Z, Huanqiang Z, Shengcai L, Zhen L, Canhui C, Lixin Z (2018) Deep hybrid similarity learning for person Re-Identification. IEEE Trans Circ Syst Video Technol 28(11):3183–3193. https://doi.org/10.1109/tcsvt.2017.2734740
44. Xiaoyan Z, Yu W, Yingbini L, Yonghui T, Guangtao W, Qinbao S (2019) A new unsupervised feature selection algorithm using similarity-based feature clustering. Comput Intell 35:2–22. https://doi.org/10.1111/coin.12192
45. Gong C, Ceyuan Y, Xiwen Y, Lei G, Junwei H (2018) When Deep Learning Meets Metric Learning: Remote Sensing Image Scene Classification via Learning Discriminative CNNs. IEEE Trans Geosci Remote Sens 56(5):2811–2821. https://doi.org/10.1109/tgrs.2017.2783902
46. Chuan-Xian R, Xiao-Lin X, Zhen L (2019) A deep and structured metric learning method for robust person Re-Identification. Pattern Recogn 96:106995–107006. https://doi.org/10.1016/j.patcog.2019.106995
47. Ha KD, Cheol SB (2021) Virtual sample-based deep metric learning using discriminant analysis. Pattern Recogn 110:107643–107656. https://doi.org/10.1016/j.patcog.2020.107643
48. Xi Y, Haoyuan G, Nannan W, Bin S, Xinbo G (2020) A Novel Symmetry Driven Siamese Network for THz Concealed Object Verification. IEEE Trans Image Process 29:5447–5456. https://doi.org/10.1109/TIP.2020.2983554
49. Yibang R, Yanshan X, Zhifeng H, Bo L (2021) A nearest-neighbor search model for distance metric learning. Inf Sci 552:261–277. https://doi.org/10.1016/j.ins.2020.11.054
50. Min C, Yongxin G, Xin F, Chuanyun X, Dan Y (2018) Person Re-Identification by pose invariant deep metric learning with improved triplet loss. IEEE Access 6:68089–68095. https://doi.org/10.1109/access.2018.2879490
51. Hantao Y, Shiliang Z, Richang H, Yongdong Z, Changsheng X, Qi T (2019) Deep Representation Learning with Part Loss for Person Re-Identification. IEEE Trans Image Process 28(6):2860–2871. https://doi.org/10.1109/TIP.2019.2891888
52. Kaiming H, Xiangyu Z, Shaoqing R, Jian S Deep Residual Learning for Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, Las Vegas, pp 770–778. https://doi.org/10.1109/CVPR.2016.90
53. Andreas V, Michael W, Serge B (2016) Residual networks behave like ensembles of relatively shallow networks. In: Advances in Neural Information Processing Systems, vol 29. Curran Associates, Barcelona, pp 550–558
54. Weiyang L, Yandong W, Zhiding Y, Ming L, Bhiksha R, Le S SphereFace: Deep Hypersphere Embedding for Face Recognition. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, Honolulu, pp 6738–6746. https://doi.org/10.1109/CVPR.2017.713
55. Prannay K, Piotr T, Chen W, Aaron S, Yonglong T, Phillip I, Aaron M, Ce L, Dilip K Supervised Contrastive Learning. In: Advances in Neural Information Processing Systems, vol 33. Annual Conference on Neural Information Processing Systems 2020, virtual. Curran Associates
56. Bac N, Bernard DB (2020) Improved deep embedding learning based on stochastic symmetric triplet loss and local sampling. Neurocomputing 402:209–219. https://doi.org/10.1016/j.neucom.2020.04.062
57. Kihyuk S (2016) Improved Deep Metric Learning with Multi-class N-pair Loss Objective, vol 29. Curran Associates, Barcelona
58. David M, Camilo N, Carlos C, Martha M, Francisco H (2020) Incremental learning model inspired in Rehearsal for deep convolutional networks. Knowl-Based Syst 208:106460–106480. https://doi.org/10.1016/j.knosys.2020.106460
59. Lake BM, Salakhutdinov R, Gross J, Tenenbaum JB One shot learning of simple visual concepts. In: Proceedings of the 33th Annual Meeting of the Cognitive Science Society. cognitivesciencesociety.org, Boston, pp 2568–2573
60. Mengye R, Eleni T, Sachin R, Jake S, Kevin S, Joshua BT, Hugo L, Richard SZ (2018) Meta-learning for Semi-Supervised Few-Shot Classification. In: 6th International Conference on Learning Representations. OpenReview.net, Vancouver
61. Qizhe X, Zihang D, HE H, Thang L, Quoc L (2020) Unsupervised Data Augmentation for Consistency Training. In: Advances in Neural Information Processing Systems, vol 33. Annual Conference on Neural Information Processing Systems, virtual. Curran Associate
62. Adam S, Sergey B, Matthew B, Daan W (2016) Timothy PL Meta-Learning with Memory-Augmented Neural Networks. In: Proceedings of the 33nd International Conference on Machine Learning. JMLR.org, New York, pp 1842–1850
63. James R, Jonathan G, John B, Sebastian N (2019) Richard ET fast and flexible Multi-Task classification using conditional neural adaptive processes. In: Advances in Neural Information Processing Systems, vol 32. PMLR, Vancouver, pp 7957–7968
64. Łukasz K, Ofir N, Aurko R, Samy B (2017) Learning to Remember Rare Events. In: 5th International Conference on Learning Representations. OpenReview.net, Toulon
65. Harrison E (2017) Neural Statistician. In: 5th International Conference on Learning Representations, Palais des Congrés NeptuneOpenReview.net, Toulon

66. Tsendsuren M, Hong Y (2017) Meta Networks. In: Proceedings of the 34th International Conference on Machine Learning. PMLR, Sydney, pp 2554–2563
67. WR F (2008) Wilcoxon Signed-Rank test. Wiley encyclopedia of clinical trials, pp 1–3. https://doi.org/10.1002/9780471462422.eoct979
68. Wenbin L, Jinglin X, Jing H, Lei W, Yang G, Jiebo L (2019) Distribution Consistency Based Covariance Metric Networks for Few-Shot Learning. In: The Thirty-Third AAAI Conference on Artificial Intelligence. AAAI, Palo Alto, pp 8642–8649. https://doi.org/10.1609/aaai.v33i01.33018642