Exploring Category-correlated Feature for Few-shot Image Classification

Jing Xu\textsuperscript{1}, Xinglin Pan\textsuperscript{2}, Xu Luo\textsuperscript{2}, Wenjie Pei\textsuperscript{1}, Zenglin Xu\textsuperscript{1}
\textsuperscript{1}Harbin Institute of Technology, Shenzhen
\textsuperscript{2}University of Electronic Science and Technology of China
xijing.may@gmail.com

Abstract

Few-shot classification aims to adapt classifiers to novel classes with a few training samples. However, the insufficiency of training data may cause a biased estimation of feature distribution in a certain class. To alleviate this problem, we present a simple yet effective feature rectification method by exploring the category correlation between novel and base classes as the prior knowledge. We explicitly capture such correlation by mapping features into a latent vector with dimension matching the number of base classes, treating it as the logarithm probability of the feature over base classes. Based on this latent vector, the rectified feature is directly constructed by a decoder, which we expect maintaining category-related information while removing other stochastic factors, and consequently being closer to its class centroid. Furthermore, by changing the temperature value in softmax, we can re-balance the feature rectification and reconstruction for better performance. Our method is generic, flexible and agnostic to any feature extractor and classifier, readily to be embedded into existing FSL approaches. Experiments verify that our method is capable of rectifying biased features, especially when the feature is far from the class centroid. The proposed approach consistently obtains considerable performance gains on three widely used benchmarks, evaluated with different backbones and classifiers. The code will be made public.

1. Introduction

Relying on massive labeled image data, deep networks, especially Convolutional Neural Networks (CNNs) [13, 15, 29] have achieved great success in image classification tasks. However, the need of large-scale data for training cannot be always satisfied when the data itself is hard to obtain (e.g., medical images, satellite images, or new species of insects) or the labor needed for labeling is prohibitive to afford. Consequentially, when not provided with enough amounts of data, deep networks like CNNs suffer from severe deterioration of generalization ability. To address such drawbacks, Few-shot Learning (FSL) [7, 30, 32] has been resurrected recently as a more practical and challenging task, aiming at adapting the classifier learned from a relatively large base dataset to novel classes of images with extremely scarce labeled data.

In real-world applications of Few-shot learning, prior knowledge of a new category is not always accessible (we...
even do not know its name). On the contrary, we have enough knowledge of base classes with their abundant labeled images. This source of knowledge is always underutilized, used only for the construction of the feature extractor and abandoned after training. A given novel category may exhibit similar patterns with one or few base classes, which could potentially help classifier construct a clearer recognition of the class; see Figure 1 for an illustrative example. This inspires us to make use of category correlations of each novel feature as a pseudo-prior information for constructing a rectified feature that is less biased and more representative of the whole class, i.e., more close to the class centroid.

To this end, we propose Category-correlated Feature Corrector (CCF), a method that augments a given novel feature based on its category correlation with base classes, generating a rectified novel feature. To be specific, the CCF is an autoencoder that explicitly models category correlation in the latent vector. This is done by matching the dimension of the latent vector with the number of base classes and directly treating them as a measure of similarity of the feature with base classes. To achieve this, in addition to the common reconstruction loss used in autoencoders, a base-class classification loss is added on the latent vector, requiring that a softmax function on the latent vector gives a proper probability of the feature over all base classes. In this way, the latent vector is restricted to encode more semantic category-related information, and discard some category-irrelevant factors which do not change the class probability of the feature. This essentially pushes the generated feature closer to class centroid. In our experiments, this property of our model is not only observed in training, but also generalizes to novel category at testing. Given a novel image, as shown in Figure 2a, the correlations with base classes can help to generate rectified feature to reduce estimation bias of the category distribution.

Additionally, we find that it is important to balance the amount of category-irrelevant information in the latent space. A too strong classification loss results in a over-clustered feature space and the generated feature deviates too much from the original one; whereas mere reconstruction is not capable of suppressing useless intra-class information. We analyse and reformulate the cross-entropy loss, finding that the temperature parameter of the softmax function plays a key role in balancing the information. Thus we carefully select its value to reach a better trade-off. At evaluation, By combining each of the original and rectified support features together, we obtain a rectified decision boundary that is a more accurate approximation of the ground truth distribution compared with that uses the biased original ones only, as shown in Figure 2b.

We validate our model by performing extensive experiments on multiple benchmark datasets. Our model achieves stable improvement on three widely used benchmarks and sets new records on mini-ImageNet. More importantly, our method is generic, flexible and agnostic to any feature extractor and classifier, readily to be embedded into existing FSL approaches. This is verified by experiments of our model applied on different feature extractors and classifiers with stable $2\% - 9\%$ improvement on 5-way 1-shot task.

### 2. Related Works

Few-shot learning task aims to recognize novel categories with limited labeled data, while given abundant training examples for the base classes. To overcome the data efficiency issue, optimization based and metric learning based methods have been proposed. Optimization based methods tackle the few shot problem by “learning to learn”. MAML [7, 8], Reptile [22] and LEO [27] aim to learn good model initialization so that the model can achieve rapid adaption on novel classes with a limited number of labeled examples. Metric learning methods attempt to map the images into a high dimensional embedding space and perform classification by comparing the distance to the representatives of each novel class, e.g., MatchingNet [32], ProtoNet [30] and Relation Network [31]. Recently,
Some papers [4, 6] reveal that the standard transfer learning paradigm can achieve surprisingly competitive performance in Few-shot learning. The paper [18] introduces a negative margin loss to reduce the discriminability on base classes and novel classes. Inspired by some newly proposed self-supervised learning methods, MoCo [12], SimCLR [3] and Byol [10], S2M2 [20], Inv-Equ [25] and ArL [41] propose to use the self-supervised learning with some regularization technique to learn high-quality feature representation. Those works demonstrate that the significance of powerful feature representations and build new strong baselines in few-shot learning tasks. Since the proposed CCF is agnostic to the feature extractors, it is easy to combine our model with the existing methods.

Another line of algorithms is generated based methods, which directly deal with data deficiency problem in novel classes. This class of methods generate novel samples by data augmentation methods [33], [11] utilizes a generator to extract transferable intra-class deformations between same-class pairs in training classes and use those information to synthesize samples efficiently. With a common starting point, [28] and [9] utilize the Auto-Encoder(AE) and Generative Adversarial Network(GAN) which makes the training process more stable. DC [38] proposes to use the statistics of the k-nearest base classes to calibrate the novel data distribution while we utilize correlation information among all the base categories. Besides, the feature generated by those methods are distributed randomly while our method can generate rectified features which are more closer to its class centers. Other methods [5, 35, 36] use high level semantic relationships between classes as extra prior information, i.e., class-level attributes [39], word2vec [21, 41] or unlabeled query data [19] to generate novel samples. However, these models are not applicable to the situation when prior knowledge is not available. Xue et al. [37] attempt to address the problem by learning a mapping function from samples to their centroids for feature rectification. However, learning the mapping without any prior knowledge is difficult and their improvement is limited to one-shot learning task. In our method, the category information can be learned automatically without introducing additional information.

3. Method

In this work, we aim to generate rectified features of novel classes conditional on the “bottleneck” latent space features to make the generated features closer to the class centers. The bottleneck latent space not only contains effective compact representation of the distribution of features, but also the corresponding class information, the category correlations between novel classes and base classes. To this end, we learn the latent space by training an Auto-Encoder on base classes and injecting the class semantic correlations of base categories into the space.

During the evaluation process, given a novel image, firstly, we use Box-Cox [2] transformation to refine the feature distribution. The Box-Cox can improve the symmetry and the normality of features and reduce the anomalies caused by the disparities of base and novel class domains. Compared with [38] using Tukey to reduce the skewness of distributions, Box-Cox is more general and can be used when negative values exist. Second, the novel feature $x_n$ is fed into an encoder, and latent feature $z$ can be seen as the category correlation among base classes. Based on the $z$, we can generate the rectified feature which is closer to its class center in the feature space. Lastly, a prediction classifier is learned based on the original feature and the generated one together. Our proposed CCF can generate rectified high-quality features to get better decision boundaries.

3.1. Problem Formulation

In Few Shot learning, we are given the base categories $C_{\text{base}}$ and novel categories $C_{\text{novel}}$, where the sets of classes $C_{\text{base}}$ and $C_{\text{novel}}$ are disjoint, $C_{\text{base}} \cap C_{\text{novel}} = \emptyset$. The base categories $C_{\text{base}}$ have abundant labeled data, $D_{\text{base}} = \{(x_i^b, y_i^b)\}_{i=0}^{m_{\text{base}}}$, where $y_i^b \in C_{\text{base}}$ and the novel categories $C_{\text{novel}}$ have scarce labeled data $D_{\text{novel}} = \{(x_i^n, y_i^n)\}_{i=0}^{m_{\text{novel}}}$, where $y_i^n \in C_{\text{novel}}$.

The common way to build the FSL task is called $N$-way $K$-shot which needs to classify $N$ classes sampled from the novel set with $K$ labeled data in each class. The few labeled data are called the support set $S = \{(x_i^n, y_i^n)\}_{i=0}^{N \times K}$ contains $N \times K$ labeled samples totally. The performance of models is evaluated on the query set $Q = \{(x_i^n, y_i^n)\}_{i=0}^{N \times K \times Q}$, where the $Q$ denotes the number of images for each class in query set, which is sampled from same $N$ classes in each episode.

Our main purpose is to predict the category of the unlabeled sample in the query set $Q$, based on the labeled support set $S$ and the auxiliary dataset $D_{\text{base}}$. In this work, our approach is over feature level and is independent of pre-trained backbones.

3.2. Category-associative Feature Corrector

We propose to use a feature generator conditional on category correlation to get rectified novel class features to obtain a better decision boundary. We train an Auto-Encoder on the base categories in the training stage and generate new novel features in the evaluation stage.

Our proposed CCF is composed of an encoder and a decoder, while adding a cross-entropy loss for classification on the latent vector, as shown in Figure 3. The CCF is a deep generative model which is effective in learning compact representation (the latent feature, $z$) of features with useful properties. Using the compact intermediate information, the mean square error(MSE) is used for feature reconstruction, which aims to generate feature, $\hat{x}$, to recovery the
input, \( x \), as much as possible. On the other way, the latent vector can be seen as category correlation, the logits before softmax function. The cross-entropy (CE) classification loss for rectification injects more category-level information into the latent vector, which makes the generated \( \hat{x} \) closer to the class center. By finding a balance between the MSE and CE losses, the CCF can generate high-quality rectified features which can diminish the the intra-class bias and help to learn better decision boundary.

### 3.2.1 Training Stage

During the training stage, our CCF is trained on the features of base classes with sufficient labeled and capture the correlation among base classes. An encoder maps an input feature \( x \in D_{\text{base}} \) into a latent variable \( z \in \mathbb{R}^{|C_{\text{base}}|} \). The dimension of \( z \) is the number of categories in base classes, \( |C_{\text{base}}| \). A decoder aims to reconstruct the feature \( \hat{x} \) from the variable \( z \). It has the form:

\[
\begin{align*}
    z &= f(x) \\
    \hat{x} &= g(z).
\end{align*}
\]

The function \( f \) and \( g \) denote the encoder and decoder, respectively. The MSE loss of CCF for reconstruction can be formulated as:

\[
\mathcal{L}_{\text{MSE}} = \sum_{x \in D_{\text{base}}} ||x - g(f(x))||^2. \tag{2}
\]

The latent space \( z \) captures the underlying generating distribution of base classes features benefited from the reconstruction MSE loss. Meanwhile, the encoder \( f \) also can be seen as a classifier and \( z \) is the logits output (the inputs to the final softmax activation for classification). The soft label \( \hat{y} \) capturing the correlation among base classes is converted by the logits, computed for each class into a probability, \( \hat{y}_i = p(y_{b=i}|x), i \in \{1, 2, ..., |C_{\text{base}}|\} \), by comparing \( z_i \) with the other logits. Compared with probability correlation \( \hat{y} \), the log probability values \( z \) contains additional fine detailed relationships between classes [1]. The softmax with temperature \( T \) can be formulated as:

\[
\hat{y}_i = \text{softmax}(z_i/T) = \frac{e^{(z_i/T)}}{\sum_j e^{(z_j/T)}}. \tag{3}
\]

The \( \hat{y} \) is supervised by the one-hot label \( y \) by the CE loss:

\[
\mathcal{L}_{\text{CE}} = \sum_{x \in D_{\text{base}}} \sum_i y_i \log(\text{softmax}(f_i(x)/T)). \tag{4}
\]

The CE penalty term for feature rectification models latent space as category correlation which maintains category-related information while removes other stochastic factors. Then, the overall loss function is defined as:

\[
\mathcal{L} = \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{CE}} + \beta ||z||^2, \tag{5}
\]

where the Frobenius norm penalty on \( z, ||z||^2 = \sum_i z_i^2 \), can avoid over-fitting on base classes features. The hyper-parameter \( \beta \) controls the strength of the regularization.

### 3.2.2 Evaluation Stage

During the evaluation stage, given an unseen feature in support set, \( x^*_i \in S \), we can get the logits feature \( z \), which can be seen as the class similarity with base categories. Based on the \( z \), the decoder generates new samples \( \hat{x}^*_i \). Since the latent feature \( \hat{z} \) discard some category-irrelevant information, the generated features \( \hat{x} \) can be rectified closer to their class centers. We train linear classifiers by combining the original support features, \( S = \{(x_i^*, y_i^*)\}_{i=0}^{N \times K} \), and the generated ones, \( S = \{(\hat{x}_n^i, y_n^i)\}_{i=0}^{N \times K} \).

### 3.3. Re-balance the Rectification and Reconstruction

The generated features by our model not only have a similar structure to the original ones (reconstruction) but also absorb more category-related information (rectification). However, having the two properties simultaneously is regarded as a pair of contradictory. We can achieve better performance by re-balancing the two properties in generated features. To this end, the cross-entropy loss can be reformulated as follows:

\[
L_{\text{CE}} = H(y, \hat{y}) = KL(y||\hat{y}) - H(y) = KL(y||\exp(z/T)\sum_j \exp(z_j/T)) - H(y) \tag{6}
\]
The features cluster tighter.

Using a higher value for rectification and reconstruction. In Figure 4 and 5, we show that we can change the hyper-parameter as possible while maintaining reconstruction performance.

The reconstruction error in generated features. As shown in Figure 4, the temperature \( T \) makes the reconstructed features absorb more category-related information. As shown in Figure 4, the latent vectors are more dispersed without CE loss compared with adding CE loss. The temperature \( T \) controls the smoothness of the distribution of \( z \). The higher value of \( T \) makes more class information absorbed in \( z \).

The mean square error loss can be rewritten as:

\[
L_{MSE} = E_x[l_2(x, g(z))] \tag{7}
\]

where the \( l_2(x, g(z)) \) denotes the \( l_2 \) distance between \( x \) and \( g(z) \). Minimizing the MSE loss keeps the \( z \) contains the distribution of the features in the latent space. However, injecting category-related information into \( z \) will lead to reconstruction error in generated features. As shown in Figure 5, the higher \( T \) with more class information results in higher reconstruction error.

Our main purpose is to rectify original feature as much as possible while maintaining the reconstruction performance. We can change the hyper-parameter \( T \) to control the class information in \( z \) and re-balance the degree between feature rectification and reconstruction. In Figure 4 and 5, we show how the two information changes when using different \( T \). Using a higher value for \( T \) makes the latent vector contain more class information and the model concentrates on the feature rectification. By decreasing the value, more attention has been turned to reconstruction.

4. Experiment

4.1. Implementation Details and Datasets

Implementation Details. The encoder of the CCF is a one-layer fully-connected network of hidden dimension 2048, varied latent dimension of \( z \) depending on the number of base categories and input dimension depending on the dimension of the pre-trained feature extractor. The decoder is a single-layer fully-connected network, which takes input as latent feature \( z \), and outputs reconstructed/generated features. The nonlinear activation function LeakyReLU is consistently used in both encoder and decoder. The whole structure is trained using Adam optimizer with a learning rate of 0.0001 with early stopping based on the Few-Shot classification accuracy on the validation set. The \( \beta \) of Frobenius norm on \( z \) is set to be 0.05 on mini/tiered-ImageNet and 0.1 on CUB. We set the temperature \( T = 0.1 \) for 1-shot task and \( T = 0.02 \) for 5-shot task. For each evaluation task, the trained CCF generates one new rectified feature for each novel support feature, followed by training a simple classifier on all support and generated features to learn a reliable decision boundary of different classes for predictions of query samples. We report mean accuracy as well as the 95% confidence interval on 2000 randomly generated episodes.

Datasets. We evaluate our method on three widely used datasets in FSL: miniImageNet [23], tieredImageNet [24] and CUB [34].

miniImageNet is a subset of ILSVRC-12 [26], including 600 images from each of 100 distinct classes of animals or objects. The categories are split into 64, 16, 20 classes for training, validation and evaluation respectively, following previous work [23].

tieredImageNet is also a subset derived from ILSVRC-12 but is much larger and is more challenging. It is made up of 779,165 images from 608 classes sampled from a hierarchical category structure. We adopt 351 classes as base categories, 97 as validation categories and 160 as novel cat-
Table 1. Average 5-way 1-shot and 5-shot classification accuracy on miniImageNet and CUB datasets. The optimization-based methods are marked as type O, metric-based as M and generated-based as G. The top results are marked in bold. The S2M2* denotes the model re-implemented by ourselves.

| Methods               | Type | miniImageNet 5-way 1-shot | miniImageNet 5-way 5-shot | CUB 5-way 1-shot | CUB 5-way 5-shot |
|-----------------------|------|---------------------------|---------------------------|------------------|------------------|
| MAML [7]              | O    | 54.69 ± 0.89              | 66.62 ± 0.83              | 51.67 ± 1.81     | 70.30 ± 0.08     |
| LEO [27]              | O    | 61.76 ± 0.08              | 77.59 ± 0.12              |                  |                  |
| Matching Net [32]     | M    | 63.08 ± 0.80              | 75.99 ± 0.60              | 54.48 ± 0.93     | 71.32 ± 0.78     |
| Prototypical Net [30] | M    | 60.37 ± 0.83              | 78.02 ± 0.57              |                  |                  |
| RelationNet [43]      | M    | 50.44 ± 0.83              | 70.20 ± 0.66              |                  |                  |
| Negative-Cosine [18]  | M    | 62.33 ± 0.82              | 80.94 ± 0.59              |                  |                  |
| CAN [14]              | M    | 64.42 ± 0.84              | 79.03 ± 0.58              |                  |                  |
| DeepEMD [40]          | M    | 65.91 ± 0.82              | 82.41 ± 0.56              |                  |                  |
| CSEI [17]             | M    | 67.59 ± 0.83              | 81.93 ± 0.36              |                  |                  |
| ARL [41]              | M    | 65.21 ± 0.58              | 80.41 ± 0.49              |                  |                  |
| Inv-Equa [25]         | M    | 67.28 ± 0.80              | 84.75 ± 0.52              |                  |                  |
| MetaGAN [42]          | G    | 52.71 ± 0.64              | 68.63 ± 0.67              |                  |                  |
| Delta-Encoder [28]    | G    | 59.9                      | 69.7                      | 69.8 ± 0.46      | 84.10 ± 0.35     |
| TriNet [5]            | G    | 58.12 ± 1.37              | 76.92 ± 0.69              | 79.56 ± 0.87     | 90.67 ± 0.35     |
| DC [38]               | G    | 68.57 ± 0.55              | 82.88 ± 0.42              |                  |                  |
| Softmax classifier [2] | M    | 51.75 ± 0.80              | 74.27 ± 0.63              | 65.51 ± 0.87     | 82.85 ± 0.55     |
| Softmax classifier*   | M    | 51.19 ± 0.43              | 73.77 ± 0.28              | 65.30 ± 0.39     | 82.63 ± 0.30     |
| Softmax classifier* + Ours | G   | **57.25 ± 0.43**          | **75.50 ± 0.29**          | **67.52 ± 0.41** | **83.55 ± 0.31** |
| Cosine classifier [4]  | M    | 51.87 ± 0.77              | 75.68 ± 0.63              | 67.02 ± 0.90     | 83.58 ± 0.54     |
| Cosine classifier*    | M    | 52.26 ± 0.43              | 75.20 ± 0.29              | 67.31 ± 0.40     | 82.98 ± 0.30     |
| Cosine classifier* + Ours | M   | **56.62 ± 0.43**          | **76.77 ± 0.30**          | **69.02 ± 0.41** | **83.79 ± 0.32** |
| S2M2 [20]             | M    | 64.93 ± 0.18              | 83.18 ± 0.11              | 80.68 ± 0.81     | 90.85 ± 0.44     |
| S2M2*                 | M    | 65.26 ± 0.43              | 83.07 ± 0.29              | 80.70 ± 0.40     | 90.70 ± 0.30     |
| S2M2* + Ours          | G    | **68.88 ± 0.43**          | **84.59 ± 0.30**          | **81.85 ± 0.42** | **91.58 ± 0.32** |

Table 2. The 5-way 1-shot and 5-shot classification accuracy on miniImageNet dataset with randomly crops 9 patches for test. The best results are marked in bold.

| Methods               | miniImageNet 5-way 1-shot | miniImageNet 5-way 5-shot |
|-----------------------|---------------------------|---------------------------|
| DeepEMD+9s [40]       | 68.77 ± 0.29              | 84.13 ± 0.53              |
| CSEI+9s [17]          | 68.94 ± 0.28              | 85.07 ± 0.50              |
| S2M2*+9s              | 69.05 ± 0.44              | 85.10 ± 0.28              |
| S2M2* + Ours + 9s     | **70.50 ± 0.44**          | **86.28 ± 0.29**          |

Table 3. 5-way (1/5)-shot classification accuracy on tieredImageNet datasets. The best results are marked in bold. The S2M2* denotes the model re-implemented by ourselves.

| Dataset              | Type | 5-way 1-shot | 5-way 5-shot |
|----------------------|------|--------------|--------------|
| MAML [7]             | O    | 51.67 ± 1.81 | 70.93 ± 0.08 |
| LEO [27]             | O    | 66.33 ± 0.05 | 81.44 ± 0.09 |
| MetaOpt [16]         | O    | 65.81 ± 0.74 | 81.75 ± 0.53 |
| CSEI [17]            | M    | 72.57 ± 0.95 | 85.72 ± 0.63 |
| DeepEMD [40]         | M    | 71.00 ± 0.32 | 85.01 ± 0.67 |
| CAN [14]             | M    | 70.65 ± 0.99 | 84.08 ± 0.68 |
| Inv-Equa [25]        | M    | 72.21 ± 0.90 | 87.08 ± 0.58 |
| S2M2*                | M    | 73.02 ± 0.49 | 86.98 ± 0.25 |
| S2M2* + Ours         | G    | **75.01 ± 0.50** | **87.65 ± 0.26** |

Table 2. The 5-way 1-shot and 5-shot classification accuracy on miniImageNet dataset with randomly crops 9 patches for test. The best results are marked in bold.

4.2. Comparisons with State-of-the-Art Methods

In this subsection, we report the results of our proposed methodology compared with other State-of-the-Art methods. We split them into 3 groups, optimization-based(O), metric-based(M), and generated-based(G) methods as introduced in the Related Works section.

The average 5-way 1/5-shot classification accuracy with 95% confidence intervals on miniImageNet and CUB datasets is suggested in [24].

CUB is a fine-grained dataset consisting of 11,788 images from 200 bird classes. Following [4], we split the dataset into 100, 50 and 50 categories as base, validation and test categories.
datasets are shown in Table 1. The miniImageNet and CUB datasets have distinct category characteristics. miniImageNet consists of a wide range of classes ranging from animals to objects with a considerably low category similarity, while in CUB there exists close correlations among fine-grained bird classes. Our proposed method can improve the performance on both miniImageNet and CUB, showing the ability to generate high-quality features and capture the category correlation with different similarity granularity. We combine our CCF with three algorithms, Softmax classifier [4], Cosine classifier [4] and S2M2 [20]. For a fair comparison, we re-implement the results of those algorithms. Compared with the baseline model S2M2, our model gives a boost to the 5-way 1/5-shot accuracy on miniImageNet by 3.63% /1.52%, and on CUB by 1.10%/0.88%, setting new records on the miniImageNet and CUB datasets. We observe similar trends on tiered-ImageNet dataset shown in Table 3.

Meanwhile, we evaluate our proposed method combined with the new sampling strategy(9s) proposed in DeepEMD [40]. Given original images, the strategy randomly samples 9 patches with different sizes and shapes followed by resizing these patches to 84 × 84. The results are shown in Table 2. We achieve remarkable 70.50% and 86.28% accuracies for 1/5-shot tasks on miniImageNet, which outperforms all the strong competitors.

4.3. Analysis

To better understand the properties of the generated features by our method, we analyze the distance from the class centers \( x_c \), to original features \( x \), and the generated ones \( \hat{x} \). Meanwhile, we use t-SNE to visualize the distribution of the generated samples. The experiments are conducted on miniImagenet.

**Distance with the class centers.** We denotes the distance between the class centers \( x_c \) with corresponding the original features \( x \) as \( d \) and with generated features as \( \hat{d} \), which \( d = ||x - x_c||_2 \) and \( \hat{d} = ||\hat{x} - x_c||_2 \). We compare the two distances in Table 4. In both base and novel classes, the distance \( \hat{d} \) has lower value than \( d \), which means the generated features are closer to the class centers. In Figure 7, we visualize the distribution of original and generated features of three randomly selected classes on base and novel classes, respectively.

Table 4. Distance with the class centers. We average all the features in both base and novel classes.

| \( l_2 \)-norm distance | base | novel |
|-------------------------|------|-------|
| \( d \)                 | 1.24 | 1.16  |
| \( \hat{d} \)           | 0.94 | 0.84  |

We attribute such advantage to the class-relevant information in latent vector \( z \) captured by our CCF. We generate new features based on the latent vector which contains class semantic correlation information and somewhat discard other information in \( z \), thereby removing some of the stochastic factors in data .

**Rectification of the outliers.** Our model works especially well due to the category gap existing between base and novel sets. It is likely that large intra-class variance exists in the novel feature space. This leads to the possibility of some novel support features being far away from its ground-truth center. We visually demonstrate in figure 7 that this problem can be largely alleviated by the rectified features from our CCF.

Figure 6. Visualization of the distribution of 3 randomly selected classes on base and novel classes, respectively. Compared with the original features(denoted as “•”), the generated features (denoted as “+”) are closer to the class centers.

Figure 7. Visualization of the distribution of some randomly generated episodes. When the support features (denoted as “★”) are far away from their ground-truth class centers, the sampled features (denoted as “×”) can rectify the outliers closer to their class centers. The query feature are denoted as “・”.
4.4. Ablation Study

In this subsection, we deeply investigate the proposed method: the effect of different components and the generalization ability of our method. All experiments are conducted on mini-ImageNet.

4.4.1 Effect of different components

Table 5. Ablation study on mini-ImageNet. The 5-way 1/5-shot classification accuracy(%) are reported.

| B  | +B-C | +AE | +CCF | 1-shot | 5-shot |
|----|------|-----|------|--------|--------|
| ✓  | ✓    | ✓   | ✓    | 65.25±0.44 | 82.85±0.29 |
| ✓  | ✓    | ✓   | ✓    | 65.97±0.43 | 83.95±0.29 |
| ✓  | ✓    | ✓   | ✓    | 67.15±0.43 | 83.89±0.28 |
| ✓  | ✓    | ✓   | ✓    | 68.88±0.43 | 84.59±0.30 |

We conduct an ablation study to assess the effects of the proposed components. We use S2M2 as the feature extractor and adopt a logistic regression classifier. The results are shown in Table 5. The first row denotes the baseline(B) that does not utilize any augmented samples for training the classifier. After using Box-Cox(+B-C) [2] to calibrate the distribution of features, we can get higher accuracy. The third line denotes that we only use the generated features to learn the classification boundary. Besides, we use the original Auto-Encoder(+AE) to generate new feature. Since the AE aims to learn a compact representation of feature and ignore insignificant data (“noise”), the accuracy is improved to some extent, but the improvement is limited. After combining the original support with the generated features(+CCF), the performance is further improved.

4.4.2 Generalization of our method

Figure 8. The histogram of 5-way 1-shot accuracies on mini-Imagenet with different backbones and classifiers.

To show our method is agnostic to different feature extractors and the classifiers, we apply our method on additional three different backbones with two few-shot classification algorithms, “Baseline” and “Baseline++” proposed in [4]. Besides, we combine three widely used classifiers: softmax, cosine and the logistic regression with our method to demonstrate the generalization ability of our method. We use the algorithm “Baseline” with softmax classifier and “Baseline++” with cosine and logistic regression classifiers. The results are shown in Figure 8. We get stable 2% – 9% on all the backbones and classifiers.

4.4.3 Hyper-parameters selection

As introduced in Section 3.3, the temperature $T$ in softmax activation is important in balancing the feature rectification and reconstruction. Here we analyze how the temperate value effect the classification performance. Figure 9 shows the results on mini-ImageNet on both 1/5-shot tasks. The best performance is achieved in 1-shot task when $T = 0.1$ and $T = 0.02$ in 5-shot task. The experiment results show that we need pay more attention for feature rectification in 1-shot task, since only one sample can lead a biased decision boundary easily. For 5-shot task, given 5 samples per class, the data-biased problem is somewhat alleviated, so, we need focus more on maintaining the original distribution of data for feature rectification.

Limitation. For $k$-shot learning, we need to balance the feature rectification and reconstruction and adjust the hyper-parameter $T$. With the increase of $k$, the improvement brought by our method will decrease gradually.

Figure 9. The accuracies of 1/5-shot task on mini-Imagenet with different temperature $T$.

5. Conclusion

In this paper, we propose a generative model named CCF for Few-shot classification. The CCF is composed of an encoder and a decoder, which can capture the latent low-dimensional manifold and transferable category correlation of features on base classes and use them to generate novel image. To be specific, the CCF is trained on base categories which encodes the feature distribution into a latent intermediate space. Meanwhile, we add a classification penalty term to encode category information into the space. Due to the penalty, the reconstructed feature outputted by the decoder is more closer to its class center, which can be seen as a rectification of the original one. After training, given a novel data, the rectified novel data is generated based on its latent encoding with the category correlation with base classes. We thoroughly verified the effectiveness of our pro-
posed methods, which can consistently achieve the appealing performance on different datasets.

References

[1] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In NIPS, pages 2654–2662, 2014.
[2] George EP Box and David R Cox. An analysis of transformations. Journal of the Royal Statistical Society: Series B (Methodological), 26(2):211–243, 1964.
[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In ICML, volume 119 of Proceedings of Machine Learning Research, pages 1597–1607. PMLR, 2020.
[4] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In ICLR (Poster). OpenReview.net, 2019.
[5] Zitian Chen, Yanwei Fu, Yinda Zhang, Xiangyang Xue, and Leonid Sigal. Multi-level semantic feature augmentation for one-shot learning. IEEE Trans. Image Process., 28(9):4594–4605, 2019.
[6] Guneet Singh Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. In ICLR. OpenReview.net, 2020.
[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In ICML, volume 70 of Proceedings of Machine Learning Research, pages 1115–1125. PMLR, 2017.
[8] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. In NeurIPS, pages 9537–9548, 2018.
[9] Hang Gao, Zheng Shou, Alireza Zareian, Hanwang Zhang, and Shih-Fu Chang. Low-shot learning via covariance-adversarial augmentation networks. In NeurIPS, pages 983–993, 2018.
[10] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheeshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In NeurIPS, 2020.
[11] Bharath Hariharan and Ross B. Girshick. Low-shot visual recognition by shrinking and hallucinating features. In ICCV, pages 3037–3046. IEEE Computer Society, 2017.
[12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In CVPR, pages 9726–9735. Computer Vision Foundation / IEEE, 2020.
[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, pages 770–778. IEEE Computer Society, 2016.
[14] Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification. In NeurIPS, pages 4005–4016, 2019.
[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, pages 1106–1114, 2012.
[16] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In CVPR, pages 10657–10665. Computer Vision Foundation / IEEE, 2019.
[17] Junjie Li, Zilei Wang, and Xiaoming Hu. Learning intact features by erasing-inpainting for few-shot classification. In AAAI, pages 8401–8409. AAAI Press, 2021.
[18] Bin Liu, Yue Cao, Yutong Lin, Qi Li, Zheng Zhang, Ming-sheng Long, and Han Hu. Negative margin matters: Understanding margin in few-shot classification. In ECCV (4), volume 12349 of Lecture Notes in Computer Science, pages 438–455. Springer, 2020.
[19] Jinhui Liu, Liang Song, and Yongqiang Qin. Prototype rectification for few-shot learning. In ECCV (1), volume 12346 of Lecture Notes in Computer Science, pages 741–756. Springer, 2020.
[20] Puneet Mangla, Mayank Singh, Abhishek Sinha, Npur Kumari, Vineeth N. Balasubramanian, and Balaji Krishnamurthy. Charting the right manifold: Manifold mixup for few-shot learning. In WACV, pages 2207–2216. IEEE, 2020.
[21] Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In NIPS, pages 3111–3119, 2013.
[22] Alex Nichol and John Schulman. Reptile: a scalable meta-learning algorithm. arXiv preprint arXiv:1803.02999, 2(3):4, 2018.
[23] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In ICLR. OpenReview.net, 2017.
[24] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-learning for semi-supervised few-shot classification. In ICLR (Poster). OpenReview.net, 2018.
[25] Mamshad Nayeem Rizve, Salman H. Khan, Fahad Shahbaz Khan, and Mubarak Shah. Exploring complementary strengths of invariant and equivariant representations for few-shot learning. In CVPR, pages 10836–10846. Computer Vision Foundation / IEEE, 2021.
[26] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. Int. J. Comput. Vis., 115(3):211–252, 2015.
[27] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In ICLR (Poster). OpenReview.net, 2019.
[28] Eli Schwartz, Leonid Karlinsky, Joseph Shitok, Sivan Harary, Mattias Marder, Abhishek Kumar, Rogério Schmidt Feris, Raja Giryes, and Alexander M. Bronstein. Delta-encoder: an effective sample synthesis method for few-shot object recognition. In NeurIPS, pages 2850–2860, 2018.
[29] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.

[30] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In NIPS, pages 4077–4087, 2017. 1, 2, 6

[31] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In CVPR, pages 1199–1208. Computer Vision Foundation / IEEE Computer Society, 2018. 1, 2

[32] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In NIPS, pages 3630–3638, 2016. 1, 2, 6

[33] Yu-Xiong Wang, Ross B. Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In CVPR, pages 7278–7286. Computer Vision Foundation / IEEE Computer Society, 2018. 3

[34] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010. 5

[35] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In CVPR, pages 5542–5551. Computer Vision Foundation / IEEE Computer Society, 2018. 3

[36] Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. F-VAEGAN-D2: A feature generating framework for any-shot learning. In CVPR, pages 10275–10284. Computer Vision Foundation / IEEE, 2019. 3

[37] Wanqi Xue and Wei Wang. One-shot image classification by learning to restore prototypes. In AAAI, pages 6558–6565. AAAI Press, 2020. 3

[38] Shuo Yang, Lu Liu, and Min Xu. Free lunch for few-shot learning: Distribution calibration. In ICLR. OpenReview.net, 2021. 3, 6

[39] Baoquan Zhang, Xutao Li, Yuning Ye, Zhichao Huang, and Lisai Zhang. Prototype completion with primitive knowledge for few-shot learning. In CVPR, pages 3754–3762. Computer Vision Foundation IEEE, 2021. 1, 3

[40] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepend: Few-shot image classification with differentiable earth mover’s distance and structured classifiers. In CVPR, pages 12200–12210. Computer Vision Foundation / IEEE, 2020. 6, 7

[41] Hongguang Zhang, Piotr Koniusz, Songlei Jian, Hongdong Li, and Philip H. S. Torr. Rethinking class relations: Absolute-relative supervised and unsupervised few-shot learning. In CVPR, pages 9432–9441. Computer Vision Foundation / IEEE, 2021. 3, 6

[42] Ruixiang Zhang, Tong Che, Zoubin Ghahramani, Yoshua Bengio, and Yangqiu Song. Metagan: An adversarial approach to few-shot learning. In NeurIPS, pages 2371–2380, 2018. 6

[43] Yueqing Zhuang, Li Tao, Fan Yang, Cong Ma, Ziwei Zhang, Huizhu Jia, and Xiaodong Xie. Relationnet: Learning deep-aligned representation for semantic image segmentation. In ICPR, pages 1506–1511. IEEE Computer Society, 2018. 6