FeatWalk: Enhancing Few-Shot Classification through Local View Leveraging

Dalong Chen[1,2], **Jianjia Zhang**[3], **Wei-Shi Zheng**[1,2], **Ruixuan Wang**[1,2,4*]

[1]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2]Key Laboratory of Machine Intelligence and Advanced Computing, MOE, Guangzhou, China
[3]School of Biomedical Engineering, Shenzhen Campus of Sun Yat-sen University, Guangdong, China
[4]Peng Cheng Laboratory, Shenzhen, China
chendlong3@mail2.sysu.edu.cn, zhangjj225@mail.sysu.edu.cn, wszheng@ieee.org, wangruix5@mail.sysu.edu.cn

Abstract

Few-shot learning is a challenging task due to the limited availability of training samples. Recent few-shot learning studies with meta-learning and simple transfer learning methods have achieved promising performance. However, the feature extractor pre-trained with the upstream dataset may neglect the extraction of certain features which could be crucial for downstream tasks. In this study, inspired by the process of human learning in few-shot tasks, where humans not only observe the whole image ('global view') but also attend to various local image regions ('local view') for a comprehensive understanding of detailed features, we propose a simple yet effective few-shot learning method called FeatWalk which can utilize the complementary nature of global and local views, therefore providing an intuitive and effective solution to the problem of insufficient local information extraction from the pre-trained feature extractor. Our method can be easily and flexibly combined with various existing methods, further enhancing few-shot learning performance. Extensive experiments on multiple benchmark datasets consistently demonstrate the effectiveness and versatility of our method. The source code is available at https://github.com/exceefind/FeatWalk.

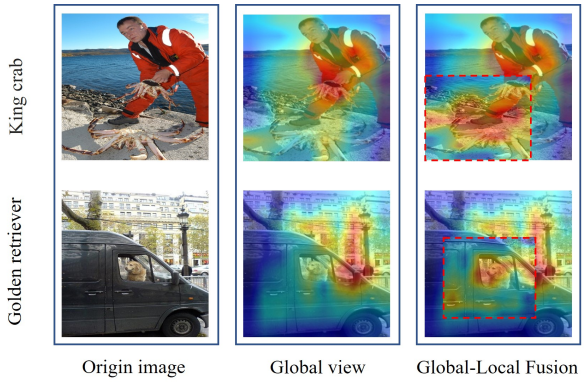

Figure 1: Two exemplar visual attentions by a representative few-shot learning method Good-Embed (Tian et al. 2020) (Middle) and our method (Right). Our method can more effectively learn to leverage local features of image patches (shown in dashed boxes) that are not well captured in the image-level features, thereby obtaining more comprehensive features.

Introduction

Convolutional neural networks (CNNs) have demonstrated excellent performance in various computer vision tasks. Its success largely relies on adequate training samples to optimize a huge number of model parameters. However, sometimes training samples may be scarce due to certain difficulty or cost in data collection. In such cases, deep neural networks often suffer from severe overfitting to limited training samples, leading to noticeable performance degradation. Therefore, how to achieve the superior model performance with very limited training samples, known as few-shot learning (FSL) (Fei-Fei, Fergus, and Perona 2006; Lake et al. 2011; Vinyals et al. 2016), has attracted strong research interest.

Recently, various approaches have been proposed to tackle the challenges of FSL. One is based on meta-learning (Finn, Abbeel, and Levine 2017; Sung et al. 2018; Vinyals et al. 2016), which simulates numerous similar FSL scenarios using an upstream dataset and trains the model to enhance its capability in learning from limited training data, and then fine-tune the model with a small amount of data to quickly adapt to the target downstream task (Jamal and Qi 2019; Lee et al. 2019; Nichol, Achiam, and Schulman 2018). Built on meta-learning of a feature extractor, metric-based methods can be employed to estimate the similarity between training and test samples for FSL (Allen et al. 2019; Li et al. 2019a; Snell, Swersky, and Zemel 2017). Meta-learning approach often requires extensive episodic training to accomplish FSL tasks. From this perspective, transfer learning approach is more desirable as simple pre-trained models can be directly applied to a wider range of target tasks. For instance, recent studies (Tian et al. 2020; Xie et al. 2022) have improved transfer learning by introducing self-distillation and enhanced embeddings, achieving significant performance gains in FSL.

In meta-learning or transfer learning, directly applying models to few-shot learning tasks may encounter some potential challenges. In learning upstream tasks, models may prioritize certain discriminative features and underestimate the importance of some other features. However, these ne-

*Corresponding author
Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

glected features could be equally or even more crucial for downstream tasks. As shown in Figure 1 (Middle column), when the feature extractor extracts features from only the whole image ('global view'), the model attends to focus on partial features and may overlook other essential information related to the current downstream task.

Inspired by the process of human learning in few-shot tasks, where humans not only observe the whole image but also attend to various local image regions ('local view') for comprehensive understanding of detailed features, we propose a simple yet effective few-shot learning method that involves sampling local views and extracting potentially important local features, thus assisting the model in gaining a more comprehensive understanding of the objects in images (Figure 1, Right column). As illustrated in Figure 2, features from multiple local views and the global view can be adaptively fused for each class with the proposed FeatWalk module (see Method section for details), such that both global features and class-relevant local features can be effectively learned from the very limited training samples in the downstream FSL task.

Our method can be flexibly combined with various meta-learning and pre-training FSL methods, without requiring changes in the meta-training or pre-training stage. With our method, new state-of-the-art performance was achieved on the standard FLS benchmark datasets *mini*ImageNet (Vinyals et al. 2016), *tiered*ImageNet (Ravi and Larochelle 2017), and CUB (Wah et al. 2011). In summary, the main contributions are as follows:

- We propose a simple yet effective FSL method that leverages local views and the FeatWalk module to obtain more comprehensive representations.
- The FeatWalk module demonstrates high flexibility, seamlessly integrating with existing FSL methods.
- Extensive experiments on benchmarks consistently validate the efficacy and adaptability of our method.

Related Works

In this section, recent studies relevant to few-shot classification are summarized, although few-shot learning has been applied to multiple types of tasks.

As one of the strategies in FSL, meta-learning aims to construct episodic training to adapt deep neural networks, particularly the feature extractor, for effective adaptation to downstream few-shot tasks (Finn, Abbeel, and Levine 2017; Li et al. 2019a; Snell, Swersky, and Zemel 2017; Vinyals et al. 2016). Most meta-learning methods are developed and evaluated under the same n-way m-shot setting, where the model is trained to predict n classes, each with only m training samples. The model is optimized through tasks that simulate downstream few-shot scenarios, enabling the feature extractor to rapidly learn from each of the n new classes. In the evaluation of meta-learning methods, after training the feature extractor on the upstream dataset, it can be fine-tuned (i.e., updated) to fast adapt to any n new classes (Finn, Abbeel, and Levine 2017; Jamal and Qi 2019; Lee and Choi 2018; Li et al. 2017; Rusu et al. 2019). Alternatively, the feature extractor can be completely fixed, and then a classifier is trained on the downstream task data for any n new classes, or classification can be performed based on metric-based methods (Hou et al. 2019; Li et al. 2019b; Ye et al. 2020a; Snell, Swersky, and Zemel 2017; Sung et al. 2018; Vinyals et al. 2016). However, meta-learning methods heavily rely on extensive episodic training.

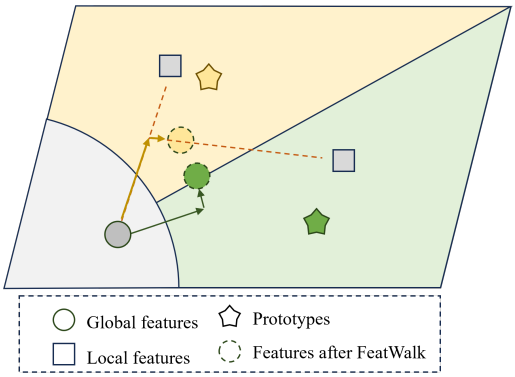

Figure 2: FeatWalk involves combining the global-view feature with multiple local-view features through weighted summation for each class. In the feature space, it can be seen as walking from the global feature (solid circles) towards a set of local representations (squares) with varying steps by presuming the local views are from a specific class (one prototype per class). As a result, fused features (dashed circles) for the corresponding classes are obtained.

Compared to various meta-learning strategies, the simple transfer learning strategy has been recently shown to achieve competitive performance on few-shot learning tasks (Chen et al. 2019; Dhillon et al. 2020; Tian et al. 2020; Xie et al. 2022). These methods don't require complicated meta-training but simple pre-training on the upstream dataset, i.e., it just needs to train a simple classifier responsible for the prediction of all classes on the upstream dataset, and then the pre-trained feature extractor is used for the downstream classification task. Most meta-training and simple transfer learning methods just learn image-level representation, and therefore certain local features crucial to downstream tasks may be filtered out if such local features are not important for the upstream task. To address this issue, some recently proposed studies (Wertheimer and Hariharan 2019; Xie et al. 2022) started to explore the way of effectively extracting local input features for FSL, e.g., by fully utilizing feature maps (Li et al. 2019a), aligning the feature distribution of both global and local views (Zhou et al. 2021), or random sampling of local features for better similarity measure (Zhang et al. 2022), etc.

Following the recent trend of utilizing local features for FSL (Hao et al. 2022; Li et al. 2020), this study provides one simple yet effective method to help the model more effectively learn to extract helpful local features with few-shot training samples for any downstream task. Unlike methods that rely on complex local similarity calculations, i.e. DeepEMD (Zhang et al. 2022), our approach simply fuses local

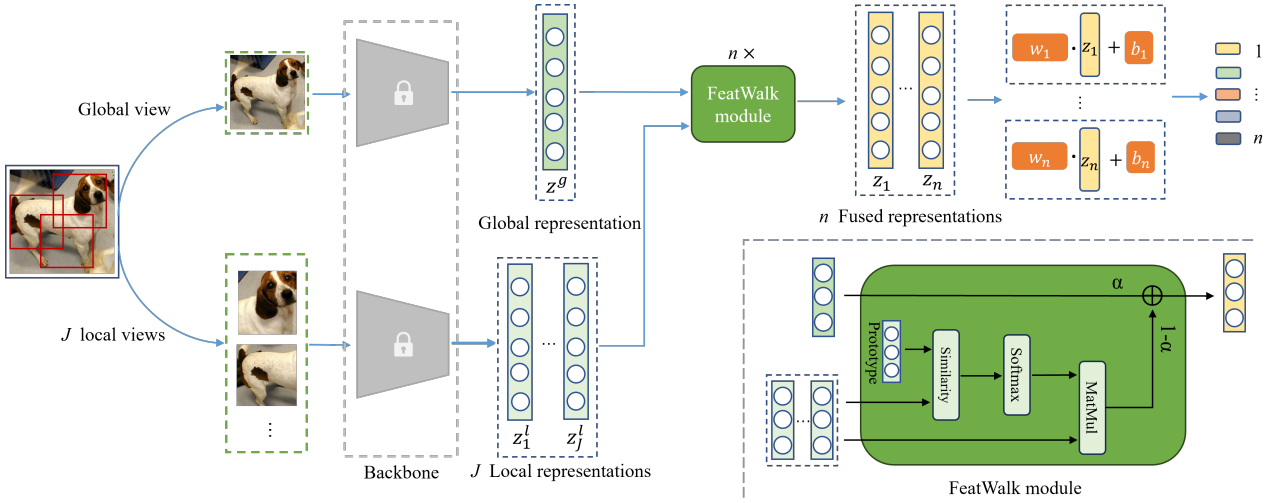

Figure 3: Overview of our method. A well-performing feature extractor backbone (Grey) is inherited from pre-training or meta-training. The fixed feature extractor is used to extract a feature vector respectively from the global view and each of the multiple local views, with the global view corresponding to the whole image and each local view corresponding to randomly sampled image patches. These feature vectors are then fed to the FeatWalk module (Lower right) to obtain a fused feature representation for each of the n classes. Each fused feature representation is then fed to the special classifier head to obtain the corresponding logit for final probability prediction. Only the classifier head (after the FeatWalk module) is learnable.

features with global features, making it more effective and straightforward to improve performance without altering the meta-learning and pre-training processes.

Method

In this section, we first introduce the potential issues in feature extraction from only the global view in few-shot learning and the motivation behind FeatWalk which aims to effectively alleviate this problem by leveraging local views. Then, we analyze how to extract and fuse global and local information for better few-shot learning. The overview of our method is demonstrated in Figure 3.

Motivation

Through sufficient episodic training in meta-learning or pre-training on large datasets, we can obtain feature extractors with exceptional representation capabilities. However, when we apply these feature extractors to few-shot learning tasks, the scarcity of samples may hinder the model from fully grasping task-relevant information. In the context of few-shot learning, humans tend to shift their attention to other local details or surrounding areas after initial global observation of the object. This cognitive process allows humans to discover potentially useful information for the current task more effectively. Similar to this process, by leveraging local perspectives and class prototypes to extract class-related information and excluding irrelevant details like backgrounds, our aim is to reveal all the potentially useful information within the limited training samples, enabling the model to make wiser decisions based on a comprehensive understanding of classes.

As depicted in Figure 1, when the feature extractor only encounters global view of images, the model may prioritize weakly related information to the current few-shot task. Consequently, without further correction from additional samples, the model might overfit to minor or irrelevant details, or even overlook crucial task-specific information. However, by additionally providing local regions as part of the input to the model, it gains a powerful ability to explore essential information that cannot be extracted from the global view alone. Hence, we believe that local views can significantly complement global views, resulting in more comprehensive and robust representations.

FeatWalk

Suppose a feature extractor $f(\cdot)$ with strong representation capability has been obtained through certain meta-learning or pre-training methods, and fixed in the downstream n-way-m-shot learning task. For any training (or test) image sample $\mathbf{x}$, we randomly sample J image patches to represent the local views. Denote by $\mathbf{z}^g$ the global representation of the sample $\mathbf{x}$, and by $\mathbf{z}_j^l$ its j-th local representation, i.e.,

$$\mathbf{z}^g = f(\mathbf{x}), \tag{1}$$

$$\mathbf{z}_j^l = f(\tilde{\mathbf{x}}_j), \tag{2}$$

where $\tilde{\mathbf{x}}_j$ is the j-th local view sampled from image $\mathbf{x}$.

With the sampled local views, we obtain multiple local representations from each image sample. However, some of these local views may only contain irrelevant information, such as that from background regions. For the classifier not to learn from such irrelevant local views, it would be desired if the importance of each local view could be appropriately estimated for class prediction of the image sample. For m training samples of one class, if one local view from one of

the m training samples is class-related (i.e., containing certain representative features of the class), there should exist similar local views in the other $(m-1)$ training samples. Otherwise, if one local view is from a background region that is irrelevant to the class of the image sample, similar local views would less likely appear in the other $(m-1)$ training samples of the same class and even may appear in training samples of other classes. Suppose there exists a prototype for each class that possesses more comprehensive and representative information of the associated class, and denote the prototype representation of the k-th class by $\mathbf{c}_k$ in the feature space. Then, the importance $\omega_{j,k}$ of the local view $\tilde{\mathbf{x}}_j$ for the k-th class can be estimated by the normalized similarity between the representation $\mathbf{z}_j^l$ of the local view and the prototype representation $\mathbf{c}_k$, i.e.,

$$\omega_{j,k} = \frac{e^{\tau \cdot s(\mathbf{z}_j^l, \mathbf{c}_k)}}{\sum_{j=1}^{J} e^{\tau \cdot s(\mathbf{z}_j^l, \mathbf{c}_k)}}, \tag{3}$$

where $s(\cdot,\cdot)$ denotes the similarity between two representations. Here we use cosine similarity, although other similarity measurements could be used as well. τ is the temperature parameter.

From Equation (3), it can be seen that local views containing class-relevant information of the k-th class as in the associated prototype will have higher importance weight $\omega_{j,k}$, while local views containing class-irrelevant information will have lower weight. Then, if the sample $\mathbf{x}$ belongs to the k-th class, the fused representation $\mathbf{z}_k$ from the global view and weighted multiple local views as follows (Equation 4, with α being the fusion coefficient) will contain more prominent features of the class compared to the only global representation $\mathbf{z}^g$, considering that the multiple local views overall contribute class-relevant features to the fused representation because class-irrelevant local representation is largely suppressed by their smaller weights,

$$\mathbf{z}_k = \alpha \mathbf{z}^g + 1 - \alpha \sum_{j=1}^{J} \omega_{j,k} \cdot \mathbf{z}_j^l \,. \tag{4}$$

In contrast, if the sample $\mathbf{x}$ is not from the k-th class, all local views would be more likely dissimilar to the prototype of the k-th class, and therefore the importance weights of all local views would be largely similar to each other. In this case, the fused representation $\mathbf{z}_k$ would probably not contain more features of the k-th class compared to the only global representation $\mathbf{z}^g$. Indeed, the fusion process (Equation 4) is equivalent to walking from the global representation $\mathbf{z}^g$ towards a set of local representations with varying steps by presuming the local views are from a specific (k-th) class. We refer to this fusion process as FeatWalk.

For a n-way-m-shot learning task, any sample $\mathbf{x}$ will result in n fused feature vectors $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n\}$ after the FeatWalk module. To make full use of all the n fused representations, a special classifier head is designed here. Specifically, let $\mathbf{w}_k$ and b_k respectively denote the learnable weight vector and the bias parameter associated with the k-th class in the classifier head, then the logit u_k for the k-th class is obtained by

$$u_k = \mathbf{w}_k \cdot \mathbf{z}_k + b_k \,, \tag{5}$$

and the logits $\{u_1, u_2, \ldots, u_n\}$ are finally sent to the softmax operator to obtain the probability output of the classifier. By default, the classifier head is trained using the traditional cross-entropy loss.

Note that the FeatWalk module requires a prototype containing representative information for each class. In this study, prototype representation $\mathbf{c}_k$ for the k-th class is initially obtained from the global representations of all the m training samples coming from the same class, i.e., $\mathbf{c}_k = \frac{1}{m}\sum_{\mathbf{x}\in D_k} \mathbf{z}^g$, and then iteratively updated based on the fused representations by $\mathbf{c}_k = \frac{1}{m}\sum_{\mathbf{x}\in D_k} \mathbf{z}_k$, where D_k is the collection of all the m training samples for the k-th class. The iterative updating of prototypes and the classifier head are performed together during training, and the updated prototypes from the last training iteration are saved for model inference. During inference, for any test image, FeatWalk is performed with the fixed prototypes, followed by the special classifier head to obtain the final output.

Experiment

Experiment Setup

Datasets: Empirical evaluations and relevant analysis were performed on the following three widely used few-shot classification datasets.

- ***Mini*ImageNet** (Vinyals et al. 2016) is a subset of the ILSVRC-12 dataset commonly used for few-shot learning, consisting of 100 classes with 600 samples per class. We used the same split as in previous studies (Ravi and Larochelle 2017) , with 64, 16, and 20 classes for training, validation, and testing, respectively.
- ***Tiered*ImageNet** (Ren et al. 2018) is a larger dataset based on the ILSVRC-12 dataset. It consists of 34 superclasses, each containing 20 sub-classes. Following the original work (Ren et al. 2018), we used 351, 97, and 160 classes for training, validation, and test, respectively.
- **CUB** (Wah et al. 2011) is a fine-grained few-shot benchmark that includes 200 classes of birds. Following previous work (Chen et al. 2019), we used 100, 50, and 50 classes for training, validation, and test, respectively.

Network Architectures: We employed various deep network structures for extensive evaluations. Specifically, we used two different backbones, ResNet-12 (Tian et al. 2020; Xie et al. 2022) and ResNet-18 (Sung et al. 2018), to enable a fair comparison with previous methods. The input resolution of images for ResNet-12 was set to 84×84 pixels, while for ResNet-18, it was set to 224×224 pixels.

Implementation Details: Our method considers the FeatWalk module as a plug-in component, making it easily applied to current few-shot learning approaches. In our method, the training of the feature extractor is consistent with the meta-learning or pre-training process of the corresponding method, e.g., ProtoNet (Snell, Swersky, and Zemel 2017) or GoodEmbed (Tian et al. 2020). Similar to

Method	Backbone	*mini*ImageNet 5-Way		*tiered*ImageNet 5-Way	
		1-shot	5-shot	1-shot	5-shot
DN4 (Li et al. 2019a)	ResNet-12	64.73 ± 0.44	79.85 ± 0.31	-	-
BML (Zhou et al. 2021)	ResNet-12	67.04 ± 0.63	83.63 ± 0.29	68.99 ± 0.50	85.49 ± 0.34
MCL (Liu et al. 2022)	ResNet-12	67.51 ± 0.20	83.99 ± 0.20	72.01 ± 0.20	86.02 ± 0.20
DeepEMD v2 (Zhang et al. 2022)	ResNet-12	68.77 ± 0.29	84.13 ± 0.53	74.29 ± 0.32	87.08 ± 0.60
CovNet (Wertheimer and Hariharan 2019)	ResNet-12	64.59 ± 0.45	82.02 ± 0.29	69.75 ± 0.52	84.21 ± 0.26
Baseline++ (Dhillon et al. 2020)	ResNet-12	60.56 ± 0.45	77.40 ± 0.34	-	-
ProtoNet (Snell, Swersky, and Zemel 2017)	ResNet-12	62.11 ± 0.44	80.77 ± 0.30	68.31 ± 0.51	83.85 ± 0.36
Good-Embed (Tian et al. 2020)	ResNet-12	64.82 ± 0.60	82.14 ± 0.43	71.52 ± 0.69	86.03 ± 0.58
DeepBDC (Xie et al. 2022)	ResNet-12	67.83 ± 0.43	85.45 ± 0.29	73.82 ± 0.47	89.00 ± 0.30
FGM (Cheng et al. 2023)	ResNet-12	69.14 ± 0.80	86.01 ± 0.62	73.21 ± 0.88	87.21 ± 0.61
RENet-ventral (Dong, Zhai, and Zha 2023)	ResNet-12	69.71 ± 0.45	84.23 ± 0.29	73.94 ± 0.48	87.15 ± 0.35
Ours	ResNet-12	**70.21 ± 0.44**	**87.38 ± 0.27**	**75.25 ± 0.48**	**89.92 ± 0.29**

Table 1: Comparison with recently proposed methods on *mini*ImageNet and *tiered*ImageNet datasets..

Method	Backbone	**CUB 5-Way**	
		1-shot	5-shot
FEAT (Ye et al. 2020b)	Conv4	68.87 ± 0.22	82.90 ± 0.15
ProtoNet (Snell, Swersky, and Zemel 2017)	Conv4	64.42 ± 0.48	81.82 ± 0.35
DeepEMD v2 (Zhang et al. 2022)	ResNet-12	79.27 ± 0.29	89.80 ± 0.51
FGM (Cheng et al. 2023)	ResNet-12	80.77 ± 0.90	92.01 ± 0.71
RENet-ventral (Dong, Zhai, and Zha 2023)	ResNet-12	83.33 ± 0.40	92.97 ± 0.24
Baseline++ (Dhillon et al. 2020)	ResNet-18	67.02 ± 0.90	83.58 ± 0.54
ADM (Li et al. 2020)	ResNet-18	79.31 ± 0.43	90.69 ± 0.21
CovNet (Wertheimer and Hariharan 2019)	ResNet-18	80.76 ± 0.42	92.05 ± 0.20
FRN (Wertheimer, Tang, and Hariharan 2021)	ResNet-18	82.55 ± 0.19	92.98 ± 0.10
Good-Embed (Tian et al. 2020)	ResNet-18	77.92 ± 0.46	89.94 ± 0.26
DeepBDC (Xie et al. 2022)	ResNet-18	84.01 ± 0.42	94.02 ± 0.24
Ours	ResNet-18	**85.67 ± 0.38**	**95.44 ± 0.16**

Table 2: Comparison with the-state-of-art methods on CUB.

these previous methods, we use the training set as the base set for meta-learning or pre-training the backbone. By default, we adopt the state-of-the-art few-shot learning method DeepBDC (Xie et al. 2022) for pre-training the feature extractor and then fix the feature extractor for subsequent FSL evaluation. During the training of the classifier head, we use the AdamW optimizer for simple and fast adaptation learning, with the learning rate of 1e-3, and optimize for 100 epochs. For each image, we randomly select 12 local views. Each local view is a 20% sized patch of the image. The temperature parameter τ is set to 32, and α is set to 0.5. To compare our method with basic and strong baseline methods, we conduct 5-way 1-shot and 5-way 5-shot classification tasks on the FSL test set, referred to as the novel set of meta-testing. For each FSL episode, we randomly select five categories (i.e., 5-way) from the test set and then, for the selected five categories, randomly choose one image (i.e., 1-shot) or five images (i.e., 5-shot) to optimize the classifier head of the framework in our method and then evaluate the classification performance using 15 query images for each category. Finally, similar to previous FSL studies (Xie et al. 2022), we conduct 2000 episodes for each run and report the average results over 5 runs with 95% confidence interval.

Comparison With State-of-the-Art Methods

General Object Recognition: As shown in Table 1, on *mini*ImageNet, our method achieves new state-of-the-art performance under both 5-way 1-shot and 5-shot settings. It outperforms the best baseline RENet-ventral, by 0.50% under the 5-way 1-shot setting and also surpasses FGM by 1.37% under the 5-way 5-shot setting. On *tiered*ImageNet, our method outperforms significantly the state-of-the-art performance under 5-way 1-shot and 5-shot settings, with accuracy improved by 0.96% and 0.92%, respectively.

Fine-Grained Categorization: Following the previous studies (Afrasiyabi, Lalonde, and Gagné 2020; Xie et al. 2022), we also evaluated our method on the fine-grained categorization task based on CUB dataset. Again, the feature extractor pre-trained by the strong baseline DeepBDC was used in our method. Table 2 reveals that our method achieves the best performance under both 5-way 1-shot and 5-shot settings, with improvements of 1.66% and 1.42%, respectively. This further confirms the effectiveness of the proposed FeatWalk as a simple transfer learning strategy in enhancing few-shot learning performance.

Cross-Domain Evaluation: To further confirm the superior transfer learning ability of our method, one cross-domain experiment was performed, i.e., training the fea-

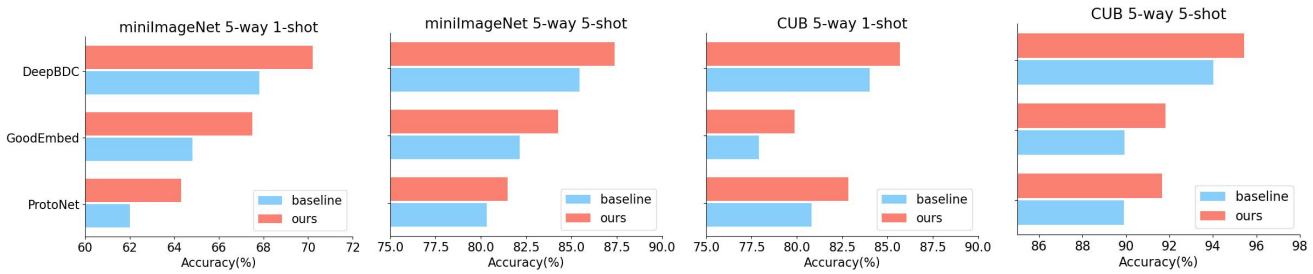

Figure 4: Performance comparison between each baseline (Blue) and its combination with our method (Red) under 5-way 1-shot and 5-shot settings on *mini*ImageNet and CUB.

Method	Backbone	5-shot
Baseline++	ResNet-18	62.04 ± 0.76
BML	ResNet-12	72.42 ± 0.54
FRN	ResNet-12	77.09 ± 0.15
ProtoNet	ResNet-12	67.19 ± 0.38
Good-Embed	ResNet-12	67.43 ± 0.44
CovNet	ResNet-12	70.55 ± 0.43
DeepBDC	ResNet-12	80.16 ± 0.38
Ours	ResNet-12	**83.60 ± 0.31**

Table 3: Performance comparison on cross-domain classification. For each method, *mini*ImageNet was used to train the feature extractor, and CUB was used for FSL evaluation.

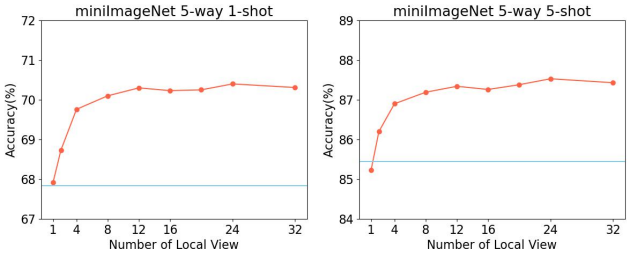

Figure 5: Sensitivity study of local view number. Blue line: performance of the corresponding strong baseline DeepBDC.

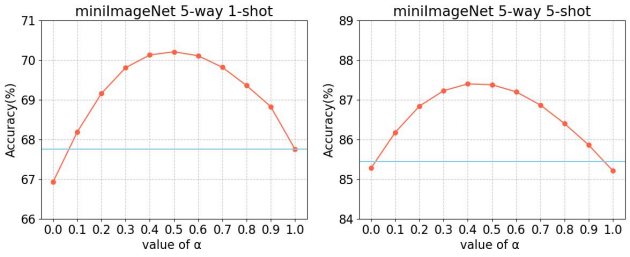

Figure 6: Sensitivity study of hyperparameter α. Blue line: performance of the strong baseline DeepBDC.

ture extractor using the general objection recognition dataset *mini*ImageNet and then transferring it to the fine-grained classification task with the CUB dataset. The cross-domain classification performance of all the baselines are directly from their original studies. As shown in Table 3, our method achieved the best cross-domain FSL performance, outperforming the best baseline DeepBDC by a large margin (83.60% vs. 80.16%). The better performance of our method is not limited to the specific DeepBDC-based feature extractor pretraining. When using the feature extractor from the Good-Embed method, our method resulted in even more significant improvement (from 67.43% to 71.45%, not shown in the table) in the cross-domain classification task.

Flexible Combinations With FSL Methods

Our method as a plug-in strategy can be flexibly combined with existing FSL methods. For fair comparison, only the FSL methods without changing the feature extractor of meta-training and pre-training are adopted, including ProtoNet, Good Embed, and DeepBDC. In this way, our method respectively used the feature extractor trained from each baseline and was compared with the corresponding baseline on both *mini*ImageNet and CUB. As Figure 4 demonstrates, introducing FeatWalk module into these method clearly improves the FSL performance compared to the corresponding baseline under both 5-way 1-shot and 5-shot settings. For example, when introducing FeatWalk module into the Good-Embed method, its performance was improved from 64.82% to 67.50% and from 82.14% to 84.25% respectively under the 5-way 1-shot and 5-way 5-shot settings on *mini*ImageNet. Note that our method is not limited to these three FSL methods. The feature extractor trained by any FSL method can be used in our method to potentially further improve the FSL performance.

Ablation Study and Sensitivity Analysis

As a plug-in strategy, our method primarily consists of random local sampling and FeatWalk operations. To investigate the impacts of these two operations, we conducted an ablation study to analyze their effects and assess their importance within the proposed method. Furthermore, we performed comparative analyses on the similarity metrics used in FeatWalk and the selection of the final classifier head. Finally, we evaluated the stability of our method with respect to the number of local view samples and the fusion coefficient between global and local representations, represented by the hyperparameter α. Further analyses about the effect

	*mini*ImageNet 5-Way		CUB 5-Way	
	1-shot	5-shot	1-shot	5-shot
FC	67.80 ± 0.45	85.21 ± 0.28	83.97 ± 0.39	94.40 ± 0.17
FC with Data Aug	68.11 ± 0.43	85.04 ± 0.28	83.04 ± 0.40	93.68 ± 0.18
FeatWalk with same view weights	69.66 ± 0.45	86.70 ± 0.27	84.91 ± 0.38	94.92 ± 0.17
FeatWalk (proposed)	**70.21 ± 0.44**	**87.38 ± 0.27**	**85.67 ± 0.38**	**95.44 ± 0.16**

Table 4: Ablation study on local views and FeatWalk module with 5-way 1-shot and 5-shot on *mini*ImageNet and CUB.

Classifier head	Measure		*mini*ImageNet 5-Way		CUB 5-Way	
	Eudist	Cosine	1 shot	5 shot	1 shot	5 shot
LR	✓		69.78±0.44	86.96±0.27	84.81±0.40	94.61±0.18
		✓	70.10±0.44	87.32±0.27	85.43±0.39	95.20±0.17
FC	✓		70.09±0.44	87.21±0.27	**85.84±0.38**	**95.51±0.16**
		✓	**70.21±0.44**	**87.38±0.27**	85.67±0.38	95.44±0.16

Table 5: Performance from different classifier heads and similarity measurements under 5-way 1-shot and 5-shot settings on *mini*ImageNet and CUB.

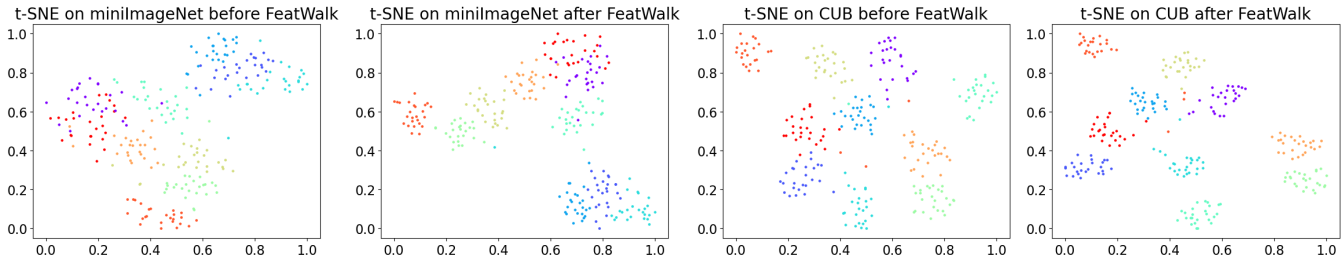

Figure 7: Visualization of features before and after applying the FeatWalk operation on *mini*ImageNet and CUB.

of the size of local view and different ways of local view acquisition are presented in Supplementary Sections A and B.

Ablation Study on FeatWalk: As shown in Table 4, the baseline method, which excludes FeatWalk module and the branch of local sampling, yields performance consistent with the reported results in the original paper (Xie et al. 2022) with fully connected layer as the classifier head (denoted as 'FC' in the first row). To certain extent, the random local sampling shares a similar concept to data augmentation. To explore whether data augmentation could improve performance, we applied data augmentation to train the classifier head with a single branch, and comparable results (second row) to the baseline were obtained, indicating the ineffectiveness of data augmentation. We further analyzed the effect of weights ($\omega_{j,k}$) for different local views in reducing the impact of irrelevant information in FeatWalk module. As Table 4 (third row) shows, when applying FeatWalk module with all local views having equal importance, the model performs better than the baseline, but it does not perform as well as the proposed method (last row). These results verify the necessity of estimating the importance of different local views and their correlation and complementarity with the global view.

Exploration of Similarity Measure and Classifier Head: In FeatWalk module, we utilized cosine similarity to analyze the importance of local representations and employed the linear transformation as in the traditional fully connected (FC) layers for the classifier head. To investigate their effects, we introduced Euclidean distance as an alternative similarity measure and explored the application of the commonly used logistic regression (LR) based classifier head (Tian et al. 2020). As shown in Table 5, the LR classifier head performs slightly less effectively than the FC-based head under the 5-way 1-shot and 5-shot settings on both *mini*ImageNet and CUB . Their performances are relatively close in terms of the two different similarity measurements.

Sensitivity Studies of Local View Numbers and FeatWalk Hyperparameters: To examine the sensitivity of the number of local view samples to the performance of our method, we gradually increased the sampling number in the 5-way 1-shot and 5-shot tasks on *mini*ImageNet . As depicted in Figure 5, the performance is quite stable when the number varies in a large range ($[12, 32]$), supporting that the proposed FeatWalk module can well utilize the class-relevant local views to help improve the few-shot learning performance when enough number of local views are available. Moreover, we conducted a sensitivity analysis on the fusion coefficient between global and local representations, represented by the hyperparameter α. Figure 6 illustrates that the model achieves the best performance within the range $[0.4, 0.6]$ for α, and the proposed method consistently achieves improved performance over the baseline in a wide range of $\alpha \in [0.1, 0.9]$, confirming that the proposed method

is insensitive to the choice of the hyperparameter α.

Visualization of Features After FeatWalk

Since our method achieves improved performance by introducing FeatWalk module, it should result in more effective representations. To better understand the changes in feature distributions before and after FeatWalk, we utilize the t-SNE technique to visualize their representations in the two-dimensional space. Figure 7 presents the feature distributions before and after applying FeatWalk to 10 new classes on both *mini*ImageNet and CUB. As observed from the visualizations, the features after applying FeatWalk within each class become more tightly clustered, and the class boundaries become clearer. Such enhanced separability between classes facilitates the following classifier head in improving classification, as confirmed by the extensive experimental evaluations in this study.

Conclusion

In this study, we proposed a simple yet effective FeatWalk module to improve the performance of few-shot learning. This module enables the feature extractor to effectively focus on extracting relevant local views for the current task. By combining local views with the global view based on the FeatWalk, our method can extract more comprehensive visual representations and therefore enhance the discriminative ability of the classifier. Our method demonstrates superior performance across multiple benchmark datasets when compared to various few-shot learning approaches. Notably, our method exhibits broad and robust applicability as it can be directly applied to the current few-shot tasks without altering the upstream meta-learning or pre-training process. The proposed method is expected to work in other data modalities such as medical images and text data, which will be investigated in future work.

Acknowledgments

This work is supported in part by the Major Key Project of PCL (grant No.PCL2023AS7-1), National Natural Science Foundation of China(grant No.62071502 and No.62101611), Guangdong Excellent Youth Team Program (grant No.2023B1515040025), Guangdong Basic and Applied Basic Research Foundation(grant No.2022A1515011375, 2023A1515012278) and Shenzhen Science and Technology Program (grant No.JCYJ20220530145411027, JCYJ20220818102414031).

References

Afrasiyabi, A.; Lalonde, J.-F.; and Gagné, C. 2020. Associative alignment for few-shot image classification. In *ECCV*.

Allen, K.; Shelhamer, E.; Shin, H.; and Tenenbaum, J. 2019. Infinite mixture prototypes for few-shot learning. In *ICML*.

Chen, W.; Liu, Y.; Kira, Z.; Wang, Y. F.; and Huang, J. 2019. A closer look at few-shot classification. In *ICLR*.

Cheng, H.; Yang, S.; Zhou, J. T.; Guo, L.; and Wen, B. 2023. Frequency Guidance Matters in Few-Shot Learning. In *ICCV*.

Dhillon, G. S.; Chaudhari, P.; Ravichandran, A.; and Soatto, S. 2020. A baseline for few-shot image classification. In *ICLR*.

Dong, L.; Zhai, W.; and Zha, Z.-J. 2023. Exploring Tuning Characteristics of Ventral Stream's Neurons for Few-Shot Image Classification. In *AAAI*.

Fei-Fei, L.; Fergus, R.; and Perona, P. 2006. One-shot learning of object categories. *PAMI*, 28(1): 594–611.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*.

Hao, F.; He, F.; Cheng, J.; and Tao, D. 2022. Global-Local Interplay in Semantic Alignment for Few-Shot Learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(7): 4351–4363.

Hou, R.; Chang, H.; Ma, B.; Shan, S.; and Chen, X. 2019. Cross attention network for few-shot classification. In *NeurIPS*.

Jamal, M. A.; and Qi, G.-J. 2019. Task agnostic meta-learning for few-shot learning. In *CVPR*.

Lake, B.; Salakhutdinov, R.; Gross, J.; and Tenenbaum, J. 2011. One shot learning of simple visual concepts. In *COGSCI*.

Lee, K.; Maji, S.; Ravichandran, A.; and Soatto, S. 2019. Meta-learning with differentiable convex optimization. In *CVPR*.

Lee, Y.; and Choi, S. 2018. Gradient-based meta-learning with learned layerwise metric and subspace. In *ICML*.

Li, W.; Wang, L.; Huo, J.; Shi, Y.; Gao, Y.; and Luo, J. 2020. Asymmetric distribution measure for few-shot learning. In *IJCAI*.

Li, W.; Wang, L.; Xu, J.; Huo, J.; Gao, Y.; and Luo, J. 2019a. Revisiting local descriptor based image-to-class measure for few-shot learning. In *CVPR*.

Li, W.; Wang, L.; Xu, J.; Huo, J.; Gao, Y.; and Luo, J. 2019b. Revisiting local descriptor based image-to-class measure for few-shot learning. In *CVPR*.

Li, Z.; Zhou, F.; Chen, F.; and Li, H. 2017. Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*.

Liu, Y.; Zhang, W.; Xiang, C.; Zheng, T.; Cai, D.; and He, X. 2022. Learning to affiliate: Mutual centralized learning for few-shot classification. In *CVPR*.

Nichol, A.; Achiam, J.; and Schulman, J. 2018. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*.

Ravi, S.; and Larochelle, H. 2017. Optimization as a model for few-shot learning. In *ICLR*.

Ren, M.; Triantafillou, E.; Ravi, S.; Snell, J.; Swersky, K.; Tenenbaum, J. B.; Larochelle, H.; and Zemel, R. S. 2018. Meta-learning for semi-supervised few-shot classification. In *ICLR*.

Rusu, A. A.; Rao, D.; Sygnowski, J.; Vinyals, O.; Pascanu, R.; Osindero, S.; and Hadsell, R. 2019. Meta-learning with latent embedding optimization. In *ICLR*.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical networks for few-shot learning. In *NeurIPS*.

Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P. H.; and Hospedales, T. M. 2018. Learning to compare: relation network for few-shot learning. In *CVPR*.

Tian, Y.; Wang, Y.; Krishnan, D.; Tenenbaum, J. B.; and Isola, P. 2020. Rethinking few-shot image classification: a good embedding is all you need? In *ECCV*.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D.; et al. 2016. Matching networks for one shot learning. In *NeurIPS*.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset.

Wertheimer, D.; and Hariharan, B. 2019. Few-shot learning with localization in realistic settings. In *CVPR*.

Wertheimer, D.; Tang, L.; and Hariharan, B. 2021. Few-shot classification with feature map reconstruction networks. In *CVPR*.

Xie, J.; Long, F.; Lv, J.; Wang, Q.; and Li, P. 2022. Joint distribution matters: deep brownian distance covariance for few-shot classification. In *CVPR*.

Ye, H.-J.; Hu, H.; Zhan, D.-C.; and Sha, F. 2020a. Few-shot learning via embedding adaptation with set-to-set functions. In *CVPR*.

Ye, H.-J.; Hu, H.; Zhan, D.-C.; and Sha, F. 2020b. Few-shot learning via embedding adaptation with set-to-set functions. In *CVPR*.

Zhang, C.; Cai, Y.; Lin, G.; and Shen, C. 2022. DeepEMD: differentiable earth mover's distance for few-shot learning. *PAMI*, 45(06): 7639–7653.

Zhou, Z.; Qiu, X.; Xie, J.; Wu, J.; and Zhang, C. 2021. Binocular mutual learning for improving few-shot classification. In *ICCV*.