SELF-SUPERVISED PRIME-DUAL NETWORKS FOR FEW-SHOT IMAGE CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We construct a prime-dual network structure for few-shot learning which establishes a commutative relationship between the support set and the query set, as well as a new self-supervision constraint for highly effective few-shot learning. Specifically, the prime network performs the forward label prediction of the query set from the support set, while the dual network performs the reverse label prediction of the support set from the query set. This forward and reserve prediction process with commutated support and query sets forms a label prediction loop and establishes a self-supervision constraint between the ground-truth labels and their predicted values. This unique constraint can be used to significantly improve the training performance of few-shot learning through coupled prime and dual network training. It can be also used as an objective function for optimization during the testing stage to refine the query label prediction results. Our extensive experimental results demonstrate that the proposed self-supervised commutative learning and optimization outperforms existing state-of-the-art few-shot learning methods by large margins on various benchmark datasets.

1 INTRODUCTION

Few-shot image classification aims to classify images from novel categories (query samples) based on very few labeled samples from each class (support images) (Hong et al., 2020a; Sun et al., 2021). During the training stage, the few-shot learning (FSL) is given a set of support-query set pairs with class labels. Once successfully trained, the model needs to be tested on unseen classes. The major challenge here is that the number of available support samples N is very small, often $N \leq 5$. In an extreme case, $N = 1$ where it is called *one-shot* learning. In order to achieve this so-called *learn-to-learn* capability, the FSL needs to capture the inherent visual or semantic relationship between the support samples and query samples, and more importantly, this learned relationship or prediction should be able to generalize well onto unseen classes (Liu et al., 2020d).

A fundamental challenge in prediction is that: if we know entity A and are trying to predict entity B, how do we know if the prediction of B, denoted by $\Phi(B)$, is accurate or not? Is there any way that we can verify the accuracy of the prediction $\Phi(B)$? As we know, this is impossible since B has no ground-truth for us to evaluate or verify its prediction accuracy. If we can come up an indirect approach to effectively evaluate the prediction accuracy, it is expected that the learning and prediction performance can be significantly improved.

In this work, we propose to explore a prime-dual commutative network design for effective prediction, specifically for few-shot image classification. As illustrated in Figure 1, the prime network Φ is the original network that learns the forward prediction from A to $\hat{B} = \Phi(A)$. The dual network Γ performs the reverse prediction from B to $\hat{A} = \Gamma(B)$. If we cascade these two networks together which establishes a prediction loop from A to B and then back to A, we have

$$\hat{A} = \Gamma(\hat{B}) = \Gamma(\Phi(A)). \tag{1}$$

Since A is given, which has the ground-truth value, the difference between A and its prime-dual loop prediction result $\hat{A}$ forms a self-supervision loss

$$\mathcal{L}_S = d(A, \hat{A}) = d(A, \Gamma(\Phi(A))), \tag{2}$$

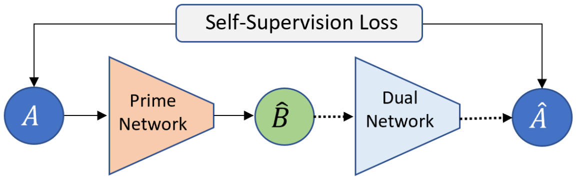


Figure 1: Illustration of the proposed idea of self-supervised prime-dual network for prediction.

where d is a distance metric function. This self-supervision loss $\mathcal{L}_S$ can be used to improve the training performance based on the coupling between the prime and dual networks. Furthermore, it can used to verify and adjust the prediction result by minimizing the self-supervision loss.

In this work, we propose to study this prime-dual network design with self-supervision for few-shot learning by exploiting the commutative relationship between the support set (entity A) and the query set (entity B). Specifically, the prime network learns to predict the labels of query samples using the support set with ground-truth labels as training samples. Meanwhile, the dual network learns to predict the labels of the support samples using the query set with ground-truth labels as training samples. For example, in 5-way 1-shot learning, the support set consists of 5 images from 5 classes with only one image per class. The query set also has 5 images from 5 classes. When training the prime and dual networks, the support set and the query set are switched for training samples. This forward and reserve prediction process with commutative support and query sets forms a label prediction loop and establishes a self-supervision constraint between the ground-truth labels and their predicted values. The prime-dual networks are jointly trained with the help from the self-supervision loss. This loss is also used during the testing stage to adjust and optimize the prediction results. Our extensive experimental results demonstrate that the proposed self-supervised commutative learning and optimization method outperforms existing state-of-the-art few-shot learning methods by a large margin on various benchmark datasets.

2 Related Work and Unique Contributions

Few-shot learning (FSL) aims to recognize instances from unseen categories with few labeled samples. There are three major categories of methods that have been developed for FSL. *(1) Data Augmentation* is the most direct method for few-shot learning, which explores different approaches to synthesize images to address the issue of few training samples. For example, self-training jigsaw augmentation (Chen et al., 2019) is able to synthesize new images by segmenting and reorganizing labeled and unlabeled gallery images. Mangla et al. (2020) apply self-supervision algorithms augmented with manifold mixup (Verma et al., 2019) for few-shot classification tasks. The F2GAN (Hong et al., 2020b) and MatchingGAN methods (Hong et al., 2020a) use generative adversarial networks (GANs) to construct high-quality samples for new image categories. *(2) Optimization-based methods* aim to learn a good initial network model for the classifier. This learned model can be then quickly adapted to novel classes using a few labeled samples. MAML (Finn et al., 2017) proposes to train a set of initialization models based on second-order gradients and meta-optimization. TAML (Jamal & Qi, 2019) reduces the bias introduced by the MAML algorithm to enforce equity between the tasks. In the Latent Embedding Optimization (LEO) method (Rusu et al., 2018), gradient-based optimization is performed in a low-dimensional latent space instead of the original high-dimensional parameter space. *(3) Metric-based methods* aim to learn a good metric space so that samples from novel categories can be effectively distinguished and correctly classified. For example, MatchingNet (Vinyals et al., 2016) applies a recurrent network to calculate the cosine similarity between samples. ProtoNet (Snell et al., 2017) compares features between samples in the Euclidean space. RelationNet (Sung et al., 2018) uses a CNN model and (Garcia & Bruna, 2017) uses the graph convolution network (GNN) to learn the metric relationship.

In this work, we also consider cross-domain FSL. For the cross-domain classification task, the model needs to generalize well from the source domain to a new or unseen target domain without accessing samples from the unseen domain during the training stage. Sun et al. (2021) propose a model-agnostic explanation-guided training method that dynamically finds and emphasizes the features which are important for the predictions. This improves the model generalization capability. To characterize the variation of image feature distribution across different domains, the LFT method (Tseng et al., 2020) learns the noise distribution by adding feature-wise transformation layers to the image encoder. To avoid over-fitting on the source domain and increase the generalization capability to the target domain, the batch spectral regularization (BSR) method (Liu et al., 2020b) attempts to suppress all singular values of the batch feature matrices during pre-training. Another set of methods (Shankar et al., 2018; Volpi et al., 2018) learn to augment the input data with adversarial learning (Yang et al., 2020b) in order to generalize the task from the source domain to the unseen target domain.

In this work, we propose a commutative prime-dual network design for few-shot learning. In the literature, the mutual dependency and reciprocal relationship between multiple modules have been explored to achieve better performance. For example, (Xu et al., 2020) has developed a reciprocal cross-task architecture for image segmentation, which improves the learning efficiency and generation accuracy by exploiting the commonalities and differences across tasks. Sun et al. (2020) design a reciprocal learning network for human trajectory prediction, which consists of forward and backward prediction neural networks. The reciprocal learning enforces consistency between the forward and backward trajectory prediction, which helps each other to improve the learning performance and achieve higher accuracy. Zhu et al. (2017) design the CycleGAN contains two GANs forming a cycle network that can translate the images of the two domains into each other to achieve style transfer. Liu et al. (2021) develop a Temporal Reciprocal Learning (TRL) approach to fully explore the discriminative information from the disentangled features. Zhang et al. (2021b) design a support-query mutual guidance architecture for few-shot object detection.

Unique Contributions. Compared to existing work in the literature, the major contributions of this work include: **(1)** We propose a new prime-dual network design to explore the commutative relationship between support and query sets and establish a unique self-supervision constraint for few-shot learning. **(2)** We incorporate the self-supervision loss into the coupled prime-dual network training to improve the few-shot learning performance. **(3)** During the test stage, using the dual network to map the prediction results back to the support set domain and using the self-supervision constraint as an objective function, we develop an optimization-based scheme to verify and optimize the performance few-shot learning. **(4)** Our proposed method has significantly advanced the state-of-the-art performance of few-shot image classification.

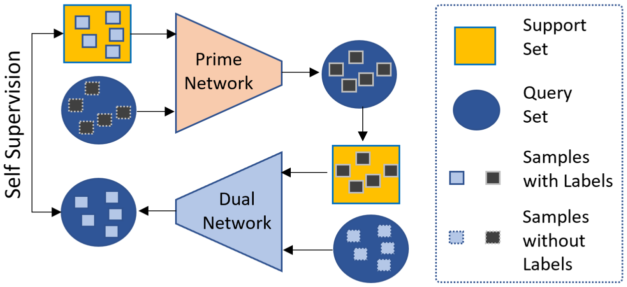


Figure 2: Illustration of the proposed idea for self-supervised prime-dual network learning and optimization for few-shot classification.

3 Method

In this section, we present our method of self-supervised prime-dual network (SPDN) learning and optimization for few-shot image classification.

3.1 Self-Supervised Commutative Learning

Figure 2 provides an overview of our proposed method of self-supervised commutative learning and optimization for few-shot image classification. In a typical setting of K-way N-shot learning, N labeled image samples from each of the K classes form the support set. For example, in a 5-way 1-shot learning, $K = 5$ and $N = 1$. Given a very small support set $\mathbf{S} = \{S_{kn} | 1 \leq k \leq K, 1 \leq n \leq N\}$, the objective of the FSL is to predict the labels of the query images $\mathbf{Q} = \{Q_{km} | 1 \leq k \leq K, 1 \leq m \leq M\}$ from the same K classes in M batches During the training stage, the labels of both support and query samples are available. The prime network $\mathbf{\Phi}_{\mathbf{S}\rightarrow\mathbf{Q}}$ for few-shot classification is trained on these support-query sets, aiming to learn and represent the inherent visual or semantic relationship between the support and query images. Once successfully learned, we will apply this network to unseen classes. Specifically, in the test stage, given a labeled support set $\mathbf{S}' = \{S'_{kn} | 1 \leq k \leq K, 1 \leq n \leq N\}$ from these K unseen classes, we need to predict the labels for the query set $\mathbf{Q}' = \{Q'_{km} | 1 \leq k \leq K, 1 \leq m \leq M\}$ also from these unseen classes.

Therefore, the fundamental challenge of FSL is to characterize and learn the inherent relationship between the support set $\mathbf{S}$ and the query set $\mathbf{Q}$. Once learned, we can then shift or transfer this relationship to $\mathbf{S}'$ and $\mathbf{Q}'$ of unseen classes to infer the labels of $\mathbf{Q}'$. In this work, as discussed in the following section, we propose to establish a graph neural network (GNN) to characterize and learn this relationship.

We recognize that, within the framework of few-shot learning, the support set and the query set are in an equal and symmetric position to each other. More specifically, if we can learn to predict the labels of query set $\mathbf{Q}$ from support set $\mathbf{S}$, certainly, we can switch their order, predicting the labels of the support set $\mathbf{S}$ from the query set $\mathbf{Q}$ using the same network architecture. This observation leads to an interesting commutative prime-dual network design for few-shot learning. As illustrated in Figure 2, we introduce a dual network $\mathbf{\Gamma_{Q \to S}}$, which performs the reverse label prediction of the support set $\mathbf{S}$ from the query set $\mathbf{Q}$. Let $\mathbf{L}(S)$ and $\mathbf{L}(Q)$ be the label vectors of $\mathbf{S}$ and $\mathbf{Q}$, respectively. Let $\hat{\mathbf{L}}(S)$ and $\hat{\mathbf{L}}(Q)$ be the predicted labels. The forward prediction by the prime network can be written as

$$\hat{\mathbf{L}}(Q) = \mathbf{\Phi_{S \to Q}}[\mathbf{L}(S)], \tag{3}$$

while the reverse prediction by the dual network can be written as

$$\hat{\mathbf{L}}(S) = \mathbf{\Gamma_{Q \to S}}[\mathbf{L}(Q)], \tag{4}$$

If both networks $\mathbf{\Phi}$ and $\mathbf{\Gamma}$ are well trained, and if we pass the label prediction output of the prime network as input to the dual network, then, we expect that the predicted labels for the support set should be close to its ground-truth. This leads to the following self-supervision loss

$$\begin{aligned} \mathcal{L}_{SS} &= ||\mathbf{L}(S) - \hat{\mathbf{L}}(S)||_2 \\ &= ||\mathbf{L}(S) - \mathbf{\Gamma_{Q \to S}}[\hat{\mathbf{L}}(Q)] \; ||_2 \\ &= ||\mathbf{L}(S) - \mathbf{\Gamma_{Q \to S}}[\mathbf{\Phi_{S \to Q}}[\mathbf{L}(S)]] \; ||_2. \end{aligned} \tag{5}$$

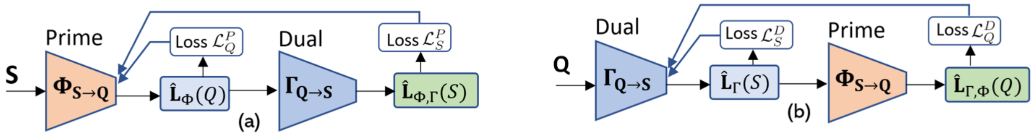

Prime
S
$\mathbf{\Phi_{S \to Q}}$
Loss $\mathcal{L}_Q^P$
$\hat{\mathbf{L}}_\Phi(Q)$
Dual
$\mathbf{\Gamma_{Q \to S}}$
Loss $\mathcal{L}_S^P$
$\hat{\mathbf{L}}_{\Phi,\Gamma}(S)$
(a)
Dual
Q
$\mathbf{\Gamma_{Q \to S}}$
Loss $\mathcal{L}_S^D$
$\hat{\mathbf{L}}_\Gamma(S)$
Prime
$\mathbf{\Phi_{S \to Q}}$
Loss $\mathcal{L}_Q^D$
$\hat{\mathbf{L}}_{\Gamma,\Phi}(Q)$
(b)

Figure 3: Illustration of the proposed idea for self-supervised commutative learning and optimization for few-shot classification.

This self-supervision constraint can be established on both support set and query set, resulting in a coupled prime-dual network training. Figure 3 (a) and (b) shows the training processes for the prime network and the dual network, respectively. Specifically, from the support set $\mathbf{S}$, the prime network learns to predict the labels of the query set $\mathbf{Q}$. As in existing few-shot learning, we have the loss $\mathcal{L}_Q^P = ||\hat{\mathbf{L}}(Q) - \mathbf{L}(Q)||_2$ between the predicted query labels and their ground-truth values. We then use the query samples and their predicted labels as input to the dual network $\mathbf{\Gamma_{Q \to S}}$, we can predict the labels of the support set $\hat{\mathbf{L}}(S)$ and compute the self-supervision loss $\mathcal{L}_S^P = ||\hat{\mathbf{L}}(S) - \mathbf{L}(S)||_2$. These two losses are combined to form the loss function for training the prime network

$$\mathcal{L}^P = ||\hat{\mathbf{L}}_\Phi(Q) - \mathbf{L}(Q)||_2 + \alpha \cdot ||\hat{\mathbf{L}}_{\Phi,\Gamma}(S) - \mathbf{L}(S)||_2. \tag{6}$$

α is a weighting parameter whose default value is set to be 0.5 in our experiments. Similarly, for the training of the dual network, as shown in Figure 3(b), its loss function is given by

$$\mathcal{L}^D = ||\hat{\mathbf{L}}_\Gamma(S) - \mathbf{L}(S)||_2 + \alpha \cdot ||\hat{\mathbf{L}}_{\Gamma,\Phi}(Q) - \mathbf{L}(Q)||_2. \tag{7}$$

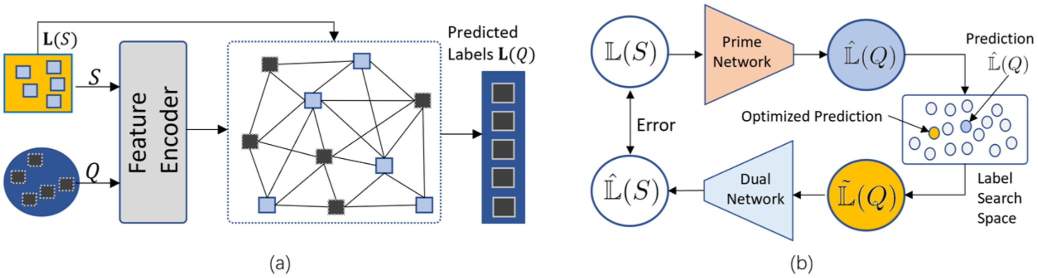


Figure 4: (a) Structure of the prime and dual network. (b) Optimizing the predicted query labels.

3.2 Graph Neural Network for Few-Shot Image Classification

The proposed prime and dual networks share the same network design, which will be discussed in this section. The only difference between these two networks is that their support and query samples are switched. In the following, we use the prime network as an example to explain its design.

The central task of few-shot learning is to characterize the inherent relationship between the query and support samples, based on which we can infer the labels of the query samples from the support samples (Tseng et al., 2020; Liu et al., 2020b). In this work, we propose to use a graph neural network (GNN) to model and analyze this relationship. In K-way N-shot learning, given K classes, each with N support samples $\{\mathbf{S}_{kn}\}$, we need to learn the prime network to predict the labels for K query samples $\{\mathbf{Q}_k\}$. This implies, in each of the total M training batch, we have $K \times (N+1)$ support samples and query samples. As illustrated in Figure 4(a), we use a backbone network, for example, Resnet-10 or Resnet-12, to extract feature for each of these support and query samples. We denote their features by $\mathbf{S} = \{\mathbf{s}_{kn}^t\}$ and $\mathbf{Q} = \{\mathbf{q}_k^t\}$ where t represents the update iteration index in the GNN. Initially, $t = 0$. These support-query sample features form the nodes for the GNN, denoted by $\{\mathbf{x}_j^t | 1 \le j \le J\}$, $J = K \times (N+1)$, for the simplicity of notations. The edge between two graph nodes represents the correlation $\psi(\mathbf{x}_i^t, \mathbf{x}_j^t)$ between nodes $\mathbf{x}_i^t$ and $\mathbf{x}_j^t$. Note that our GNN has two groups of nodes: support sample nodes and query sample nodes. The support samples nodes have labels while the labels of the query samples need to be predicted by the prime network. If $\mathbf{x}_i^t$ and $\mathbf{x}_j^t$ are both support nodes, we have

$$\psi(\mathbf{x}_i^t, \mathbf{x}_j^t) = \begin{cases} 1 & if\ \mathbf{L}(\mathbf{x}_i^t) = \mathbf{L}(\mathbf{x}_j^t), \\ 0 & if\ \mathbf{L}(\mathbf{x}_i^t) \neq \mathbf{L}(\mathbf{x}_j^t). \end{cases} \tag{8}$$

Here, $\mathbf{L}(\cdot)$ represents the label of the corresponding support sample. Since the labels for the query nodes are unknown, the correlation for edges linked to these query nodes need to be learned by the GNN. Initially, we set them to be random values between 0 and 1.

Each node of the GNN combines features from these neighboring nodes with the corresponding correlation as weights and updates its own feature by learning a multi-layer perceptron (MLP) network $\mathcal{G}_o[\cdot]$ as follows

$$\mathbf{x}_j^{t+1} = \mathcal{G}_o\left[\sum_{i=1}^{J} \mathbf{x}_j^t \cdot \psi(\mathbf{x}_i^t, \mathbf{x}_j^t)\right]. \tag{9}$$

At each edge, another MLP network $\mathcal{G}_e[\cdot,\cdot]$ is learned to predict the correlation between two graph nodes,

$$\psi(\mathbf{x}_i^t, \mathbf{x}_j^t) = \mathcal{G}_e[\mathbf{x}_i^t, \mathbf{x}_j^t], \tag{10}$$

whose ground-truth values are obtained using the scheme discussed in the above. The feature generated by the prime GNN is then passed to a classification network to predict the query labels. Both the prime and dual GNNs are jointly trained with their final classification networks.

3.3 Self-Supervised Optimization of Few-Shot Image Classification

Besides improving the training performance through mutual enforcement, the proposed self-supervised prime-dual network design can be also used in the testing stage to optimize the label prediction of query samples. Specifically, we can use the dual network to refine and optimize the label prediction results obtained by the prime network. As illustrated in Figure 4(b), given a support set $\mathbf{S}$ and a query set $\mathbf{Q}$, the support set has class labels $\mathbf{L}(S)$. Let $\hat{\mathbf{L}}(Q)$ be the prediction result, the output of the softmax layer of the classification network. In existing approaches of few-shot learning or other network prediction scenarios, we are not able to verify if the prediction is accurate or not since the ground-truth is not available for test samples. However, in this work, with the dual network $\mathbf{\Gamma}_{\mathbf{Q}\rightarrow\mathbf{S}}$ being successfully trained, we can use the prediction result $\hat{\mathbf{L}}(Q)$ as input to the dual network to predict the class labels of the original support samples

$$\hat{\mathbf{L}}(S) = \mathbf{\Gamma}_{\mathbf{Q}\rightarrow\mathbf{S}}[\hat{\mathbf{L}}(Q)]. \tag{11}$$

Note that these support samples DO have ground-truth labels $\mathbf{L}(S)$. Define the label prediction error by

$$\mathbf{E}_l(S) = ||\mathbf{L}(S) - \hat{\mathbf{L}}(S)||_2. \tag{12}$$

We assume that the correct query sample labels $\mathbf{L}^*(Q)$ is within the neighborhood of the prediction result $\hat{\mathbf{L}}(Q)$. Let $\mathbf{\Omega}$ be the set of candidate assignments of query labels which are within the neighborhood of $\hat{\mathbf{L}}(Q)$. For example,

$$\mathbf{\Omega} = \{\tilde{\mathbf{L}}(Q) : \ ||\tilde{\mathbf{L}}(Q) - \hat{\mathbf{L}}(Q)||_2 \leq \Delta\}, \tag{13}$$

where Δ is a given threshold for the label vector distance. We then search the candidate query labels $\tilde{\mathbf{L}}(Q)$ within the neighborhood set $\mathbf{\Omega}$ to minimize the support label prediction error $\mathbf{E}_l(S)$ in (12). The optimized prediction of the query samples is given by

$$\mathbf{L}^*(Q) = \arg \min_{\tilde{\mathbf{L}}(Q)\in\mathbf{\Omega}} ||\ \mathbf{L}(S) - \mathbf{\Gamma}_{\mathbf{Q}\rightarrow\mathbf{S}}[\tilde{\mathbf{L}}(Q)]\ ||_2. \tag{14}$$

From the experimental results, we will see that this unique self-supervised optimization of the query label prediction is able to significantly improve the few-shot image classification performance.

4 Experimental Results

In this section, we provide experimental results on various benchmark datasets to demonstrate the performance of our proposed SPDN method for few-shot learning.

4.1 Implementation Details

We use ResNet-10 as the backbone of our feature encoder. The input images are resized to 224×224 and the output feature vector size is $1 \times 1 \times 512$. We choose the Adam optimizer with a learning rate of 0.01 and a batch size of 64 for training of 400 epochs. In the episodic meta-training stage, we use the graph neural network (GNN) discussed in the above section to generate the feature embedding for query samples. The prime network $\mathbf{\Phi}_{\mathbf{S}\rightarrow\mathbf{Q}}$ and the dual network $\mathbf{\Gamma}_{\mathbf{Q}\rightarrow\mathbf{S}}$ are jointly trained. These two networks are both trained for 400 epochs with 100 episodes per epoch. In each episode, we randomly select K categories (K=5, 5-way) from the training set. Then, we randomly select N samples (N=1 or 5 for 1-shot or 5-shot) from each category to compose support set and query set, respectively. In the test stage, we use the average of 1000 trials as the final result for all the experiments. For each trial, we randomly select K categories from the test set. Similar to the training stage, N (1 or 5) samples are randomly selected as the support set and 15 samples as the query set from each category.

4.2 Datasets

Five benchmark datasets are used for performance evaluation and comparison with other methods in the literature, Mini-ImageNet (Ravi & Larochelle, 2016), CUB (Wah et al., 2011), Cars (Krause et al., 2013), Places (Zhou et al., 2017) and Plantae (Van Horn et al., 2018). More details about dataset settings are presented in Appendix A.1.

4.3 Results

To demonstrate the performance of our SPDN method, we conduct a series of experiments under different few-shot classification settings. In the literature, there are two major scenarios for testing the FSL methods: (1) intra-domain learning where the training classes and test classes are from the same object domain, for example, both from the Mini-ImageNet classess, and (2) cross-domain learning where the FSL is trained on one dataset (e.g., Mini-ImageNet) and the testing is performed on another dataset (e.g., CUB). Certainly, the cross-domain scenario is more challenging.

4.3.1 Intra-Domain FSL Results.

First, we conduct intra-domain FSL experiments on the Mini-ImageNet. Table 1 summarizes the performance comparison with state-of-the-art FSL methods mainly developed in the past two years. We also list the backbone network used for extracting the features for the input images. We can see that, for the 5-way 1-shot image classification task, our method (with ResNet-10 backbone) outperform the current best method (with ResNet-12 backbone) from (Zhang et al., 2021a) by 5.42%. Another method which uses the same ResNet-10 backbone is the GNN+FT method (Tseng et al., 2020). Our method outperforms this method by 12.23%. For the 5-way 5-shot classification task, our method outperforms the current best by more than 5%, which is quite significant.

Table 1: The results of general few-shot classification trained on Mini-ImageNet with 95% confidence intervals.

Methods	Backbone	Mini-ImageNet	
		1-shot	5-shot
ProtoNet (Snell et al., 2017)	ResNet-12	62.39 ± 0.21%	80.53 ± 0.14%
MetaOpNet (Lee et al., 2019)	ResNet-12	62.64 ± 0.61%	78.63 ± 0.46%
Robust 20-distill (Dvornik et al., 2019)	ResNet-18	63.06 ± 0.61%	80.63 ± 0.42%
SimpleShot (Wang et al., 2019)	ResNet-18	62.85 ± 0.20%	80.02 ± 0.14%
CAN (Hou et al., 2019)	ResNet-12	63.85 ± 0.48%	79.44 ± 0.34%
EGNN (Kim et al., 2019)	ResNet-12	–	76.37%
Meta-Baseline (Chen et al., 2020)	ResNet-12	63.17 ± 0.23%	79.26 ± 0.17%
GNN+FT (Tseng et al., 2020)	ResNet-10	66.32 ± 0.80%	81.98 ± 0.55%
FEAT (Ye et al., 2020)	ResNet-12	66.78 ± 0.20%	82.05 ± 0.14%
DeepEMD (Zhang et al., 2020)	ResNet-12	65.91 ± 0.82%	82.41 ± 0.56%
Neg-Cosine (Liu et al., 2020a)	ResNet-12	63.85 ± 0.81%	81.57 ± 0.56%
EBM (Liu et al., 2020e)	ResNet-12	64.09 ± 0.37%	80.29 ± 0.25%
RFS-distill (Tian et al., 2020)	ResNet-12	64.82 ± 0.60%	82.14 ± 0.43%
DPGN (Yang et al., 2020a)	ResNet-12	67.77 ± 0.32%	84.60 ± 0.43%
FRN (Wertheimer et al., 2021)	ResNet-12	66.45 ± 0.19%	82.83 ± 0.13%
IER-Distill (Rizve et al., 2021)	ResNet-12	67.28 ± 0.80%	84.78 ± 0.52%
Zhang et al. (Zhang et al., 2021a)	ResNet-12	73.13 ± 0.85%	82.06 ± 0.54%
COSOC (Luo et al., 2021)	ResNet-12	69.28 ± 0.49%	85.16 ± 0.42%
Our SPDN Method	ResNet-10	**78.55 ± 0.74**%	**91.10 ± 0.39**%
Performance Gain		**+5.42**%	**+5.94**%

Second, we evaluate our method on intra-domain fine-grained image classification tasks on the CUB dataset. In this case, the FSL needs to learn subtle features to distinguish objects from close categories. Table 2 summarizes the performance results on 5-way 1-shot and 5-way 5-shot classification tasks. We can see that, for the one-shot classification task, our method outperforms the current best method, FRN (Wertheimer et al., 2021) by 6.72%. For the 5-shot classification task, our method improves the classification accuracy by 2.80%.

Table 2: The results of fine-grained few-shot classification models trained on CUB.

Model	Backbone	1-shot	5-shot
ProtoNet (Snell et al., 2017)	ResNet-18	72.99 ± 0.88%	86.64 ± 0.51%
Neg-Cosine (Liu et al., 2020a)	ResNet-18	72.66 ± 0.85%	89.40 ± 0.43%
Centroid-A (Afrasiyabi et al., 2020)	ResNet-18	74.22 ± 1.09%	88.65 ± 0.55%
BD-CSPN (Liu et al., 2020c)	ResNet-18	78.89%	88.70%
LaplacianShot (Ziko et al., 2020)	ResNet-18	80.96%	88.68%
TFH (Lazarou et al., 2021)	ResNet-18	75.83%	88.17%
ProtoNet (Snell et al., 2017)	ResNet-12	78.60 ± 0.62%	89.73 ± 0.12%
DSN (Simon et al., 2020)	ResNet-12	79.96 ± 0.21%	91.41 ± 0.34%
CTX (Doersch et al., 2020)	ResNet-12	79.34 ± 0.21%	91.42 ± 0.11%
FRN (Wertheimer et al., 2021)	ResNet-12	83.55 ± 0.19%	92.92 ± 0.10%
Our SPDN Method	ResNet-12	**90.27 ± 0.48%**	**95.72 ± 0.41%**
Performance Gain		**+6.72%**	**+2.80%**

4.3.2 Cross-Domain FSL Results.

The cross-domain few-shot learning is more challenging. Following existing methods, we train the model on the Mini-ImageNet object domain and test the trained model on other domains, including the CUB, Cars, Places and Plantae datasets. Table 3 summarizes the results for 5-way 1-shot classification (top) and 5-way 5-shot classification (bottom). We can see that our SPDN method has dramatically improved the classification accuracy on these cross-domain FSL tasks. For example, on the Cars dataset, our method outperforms the current best TPN+ATA (Wang & Deng, 2021) by 4.15%. On the Plantae dataset, the performance gain is 5.59%, which is quite significant. For the 5-way 5-shot classification task, the performance gains on these datasets are also very significant, between 0.37-8.68%. This demonstrates that our SPDN method is able to learn the inherent visual relationship between the support and query samples and can generalize very well onto unseen classes in new object domains.

Table 3: Comparison of different methods on cross-domain few-shot classification, the model is trained on Mini-ImageNet and test on multiple datasets.

5-way 1-shot	CUB	Cars	Places	Plantae
GNN+FT (Tseng et al., 2020)	$47.47 \pm 0.75\%$	$31.61 \pm 0.53\%$	$55.77 \pm 0.79\%$	$35.95 \pm 0.58\%$
GNN+ATA (Wang & Deng, 2021)	$46.23 \pm 0.50\%$	$37.15 \pm 0.40\%$	$54.18 \pm 0.50\%$	$37.38 \pm 0.40\%$
LRP-GNN (Sun et al., 2021)	$48.29 \pm 0.51\%$	$32.78 \pm 0.39\%$	$54.83 \pm 0.56\%$	$37.49 \pm 0.43\%$
TPN+ATA (Wang & Deng, 2021)	$51.89 \pm 0.50\%$	$38.07 \pm 0.40\%$	$57.26 \pm 0.50\%$	$40.75 \pm 0.40\%$
Our SPDN Method	$\mathbf{57.90 \pm 0.76\%}$	$\mathbf{42.22 \pm 0.65\%}$	$\mathbf{68.73 \pm 0.84\%}$	$\mathbf{46.34 \pm 0.64\%}$
Performance Gain	$\mathbf{+6.01\%}$	$\mathbf{+4.15\%}$	$\mathbf{+11.47\%}$	$\mathbf{+5.59\%}$
5-way 5-shot	CUB	Cars	Places	Plantae
GNN+FT (Tseng et al., 2020)	$66.98 \pm 0.68\%$	$44.90 \pm 0.64\%$	$73.94 \pm 0.67\%$	$53.85 \pm 0.62\%$
GNN+ATA (Wang & Deng, 2021)	$69.83 \pm 0.50\%$	$54.28 \pm 0.40\%$	$76.64 \pm 0.40\%$	$58.08 \pm 0.40\%$
LRP-GNN (Sun et al., 2021)	$64.44 \pm 0.48\%$	$46.20 \pm 0.46\%$	$74.45 \pm 0.47\%$	$54.46 \pm 0.46\%$
TPN+ATA (Wang & Deng, 2021)	$70.14 \pm 0.40\%$	$55.23 \pm 0.40\%$	$73.87 \pm 0.40\%$	$59.02 \pm 0.40\%$
Our SPDN Method	$\mathbf{77.72 \pm 0.65\%}$	$\mathbf{55.60 \pm 0.69\%}$	$\mathbf{85.32 \pm 0.51\%}$	$\mathbf{64.46 \pm 0.62\%}$
Performance Gain	$\mathbf{+7.58\%}$	$\mathbf{+0.37\%}$	$\mathbf{+8.68\%}$	$\mathbf{+5.44\%}$

4.4 Ablation Studies

In this section, we conduct ablation studies to further understand the proposed SPDN method and analyze the contributions of major algorithm components.

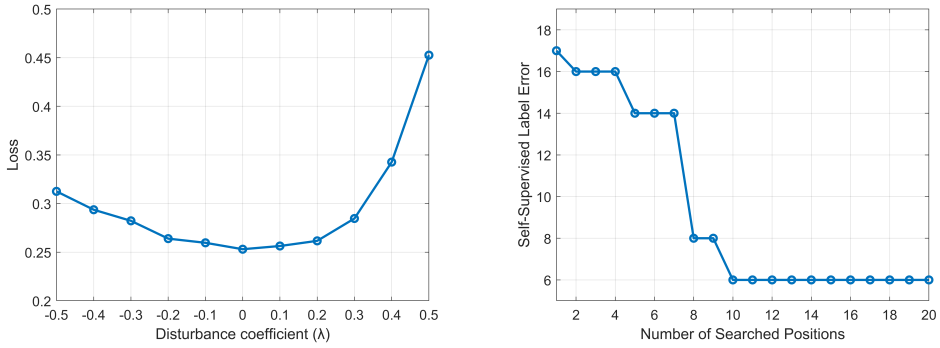
From algorithm design perspective, our SPDN method has two major components: self-supervised learning (SSL) of the prime and dual networks, and the self-supervised optimization (SSO) of the predicted query labels. We adopt the single GNN-based model (Tseng et al., 2020) as the baseline of our method and the SSL and SSO algorithm components are added onto this baseline method. To understand the performance of these two algorithm components, in the following experiment, we train the SPDN method using training samples from the Mini-ImageNet. We conduct intra-domain few-shot image classification on the Mini-ImageNet and cross-domain few-shot image classification on the CUB, Cars, Places, and Plantae datasets. Table 4 summarizes the results for 5-way 1-shot and 5-way 5-shot image classification. The second column shows the intra-domain few-shot image classification results on the Mini-ImageNet. The rest columns show the results for the cross-domain classification results. We can see that the self-supervised prime-dual network training is able to improve the classification accuracy by up to 1.8%. The performance gain achieved by the self-supervised optimization of the predicted query labels is much more significant, ranging from 7-10%. This dramatic performance gain is a surprise to us. In the following, we will provide additional ablation studies to further understand this SSO algorithm module. Compared to the SSO module, the performance improvement by the first SSL module is relatively small. This is because the major new contribution of the SSL module is the self-supervised loss which aims to further improve the learning on the baseline GNN. However, it has successfully trained a dual network, which plays a very important role in the second SSO module. It is used to search and optimize the predicted labels of the query samples, resulting in major performance gain. We discuss the specific optimization results of our self-supervised optimization (SSO) module through an experiment in Appendix A.3.

Table 4: Contributions of algorithm components, self-supervised learning of prime-dual networks and self-supervised optimization of predicted query labels.

5-way 1-shot	Mini-ImageNet	CUB	Cars	Places	Plantae
Baseline	$66.41 \pm 0.72\%$	$46.97 \pm 0.68\%$	$31.58 \pm 0.62\%$	$55.84 \pm 0.77\%$	$36.09 \pm 0.53\%$
+SSL	$67.93 \pm 0.65\%$	$46.99 \pm 0.72\%$	$33.39 \pm 0.58\%$	$58.06 \pm 0.67\%$	$36.73 \pm 0.70\%$
+SSL+SSO	$78.55 \pm 0.74\%$	$57.90 \pm 0.76\%$	$42.22 \pm 0.65\%$	$68.73 \pm 0.84\%$	$46.34 \pm 0.64\%$
5-way 5-shot	Mini-ImageNet	CUB	Cars	Places	Plantae
Baseline	$82.66 \pm 0.67\%$	$66.85 \pm 0.58\%$	$43.53 \pm 0.66\%$	$74.53 \pm 0.61\%$	$52.83 \pm 0.64\%$
+SSL	$83.21 \pm 0.52\%$	$67.71 \pm 0.71\%$	$45.17 \pm 0.56\%$	$75.63 \pm 0.62\%$	$54.67 \pm 0.66\%$
+SSL+SSO	$91.10 \pm 0.39\%$	$77.72 \pm 0.65\%$	$55.60 \pm 0.69\%$	$85.32 \pm 0.51\%$	$64.46 \pm 0.62\%$

In the following experiments, we attempt to further understand the behavior and performance of the SSO algorithm module. First, we conduct an experiment to understand the search and optimization process of SSO. Suppose $\mathbf{L}(Q)$ is the true label of the query samples. Let

$$\tilde{\mathbf{L}}(Q) = \mathbf{L}(Q) + \lambda \cdot \Delta_{\mathbf{L}}, \tag{15}$$



(a) (b)

Figure 5: (a) The self-supervised label error of support samples as a function of the disturbance coefficient λ on the predicted query labels. (b) The self-supervised label error decreases with the number of search positions.

be a label vector within the neighborhood of $\mathbf{L}(Q)$. Here, $\Delta_{\mathbf{L}}$ is a pre-generated random vector and λ is a disturbance coefficient to control the amount of variation. With the label vector $\tilde{\mathbf{L}}(Q)$ and the query samples, we can predict the labels of the support vector using the dual network. Then, we can compute the prediction error $\mathbf{E}_l(S)$ as in (12). Figure 5(a) shows the label prediction error $\mathbf{E}_l(S)$ as a function of λ. This experiment was performed on 5-way 1-shot image classification on the CUB dataset. We can see that the minimum error is achieved at $\lambda = 0$. This implies the ground-truth labels of the query samples have the minimum self-supervised label error $\mathbf{E}_l(S)$. This is a very important property of our SSO method. It suggests that, when the predicted query labels are not correct, and the ground-truth labels are within its neighborhood, we can use the SSO method to search for these ground-truth labels using the minimum self-supervised support label error criteria.

During our self-supervised optimization of the predicted query labels, we choose a small neighborhood $\mathbf{\Omega}$ within the neighborhood of the predicted query labels $\hat{\mathbf{L}}(Q)$ with a maximum distance Δ. This $\Delta controls$ the number of search positions in the label space. If we search more positions or candidate query labels, we can obtain smaller self-supervised label errors of the support samples $\mathbf{E}_l(S)$. Figure 5(b) plots the value of $\mathbf{E}_l(S)$ as a function of the number of search positions. This experiment was performed on 5-way 1-shot image classification on the CUB dataset. We can see that the error drops significantly with the number of searched positions. We recognize that, for each search position, we need to run the dual network once. This does introduce extra computational complexity. But, the amount of performance gain is very appealing. In our experiments, we limited the number of search positions to the 5, i.e., the nearest 5 label vectors (integer vectors) to the predicted query label.

5 Conclusion

In this work, we have successfully developed a novel prime-dual network structure for few-shot learning which explores the commutative relationship between the support set and the query set. The prime network performs the forward label prediction from the support set to the query set, while the dual network performs the reverse label prediction from the query set to the support set. This forward and reserve prediction process with commutative support and query sets forms a label prediction loop and establishes a self-supervision constraint between the ground-truth labels and their predicted values. We have established a self-supervised support error metric and used the learned dual network to optimize the predicted query labels during the testing stage. Our extensive experimental results on both intra-domain and cross-domain few-shot image classificaiton have demonstrated that the proposed self-supervised prime-dual network learning and optimization have significantly improved the performance of few-shot learning, especially for cross-domain few-shot learning tasks. We have also conducted detailed ablation studies to provide in-depth understanding of the significant performance gain achieved by the self-supervised optimization process. The self-supervised prime-dual network design is general and can be naturally incorporated into other prediction and learning methods.

REFERENCES

Arman Afrasiyabi, Jean-François Lalonde, and Christian Gagné. Associative alignment for few-shot image classification. In *European Conference on Computer Vision*, pp. 18–35. Springer, 2020.

Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *arXiv preprint arXiv:2003.04390*, 2020.

Zitian Chen, Yanwei Fu, Kaiyu Chen, and Yu-Gang Jiang. Image block augmentation for one-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3379–3386, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Carl Doersch, Ankush Gupta, and Andrew Zisserman. Crosstransformers: spatially-aware few-shot transfer. *arXiv preprint arXiv:2007.11498*, 2020.

Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3723–3731, 2019.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pp. 1126–1135. PMLR, 2017.

Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043*, 2017.

Yan Hong, Li Niu, Jianfu Zhang, and Liqing Zhang. Matchinggan: Matching-based few-shot image generation. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2020a.

Yan Hong, Li Niu, Jianfu Zhang, Weijie Zhao, Chen Fu, and Liqing Zhang. F2gan: Fusing-and-filling gan for few-shot image generation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 2535–2543, 2020b.

Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification. *arXiv preprint arXiv:1910.07677*, 2019.

Muhammad Abdullah Jamal and Guo-Jun Qi. Task agnostic meta-learning for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11719–11727, 2019.

Jongmin Kim, Taesup Kim, Sungwoong Kim, and Chang D Yoo. Edge-labeling graph neural network for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11–20, 2019.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 554–561, 2013.

Michalis Lazarou, Yannis Avrithis, and Tania Stathaki. Tensor feature hallucination for few-shot learning. *arXiv preprint arXiv:2106.05321*, 2021.

Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10657–10665, 2019.

Bin Liu, Yue Cao, Yutong Lin, Qi Li, Zheng Zhang, Mingsheng Long, and Han Hu. Negative margin matters: Understanding margin in few-shot classification. In *European Conference on Computer Vision*, pp. 438–455. Springer, 2020a.

Bingyu Liu, Zhen Zhao, Zhenpeng Li, Jianan Jiang, Yuhong Guo, and Jieping Ye. Feature transformation ensemble model with batch spectral regularization for cross-domain few-shot classification. *arXiv preprint arXiv:2005.08463*, 2020b.

Jinlu Liu, Liang Song, and Yongqiang Qin. Prototype rectification for few-shot learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pp. 741–756. Springer, 2020c.

Lu Liu, William Hamilton, Guodong Long, Jing Jiang, and Hugo Larochelle. A universal representation transformer layer for few-shot image classification. *arXiv preprint arXiv:2006.11702*, 2020d.

Xuehu Liu, Pingping Zhang, Chenyang Yu, Huchuan Lu, and Xiaoyun Yang. Watching you: Global-guided reciprocal learning for video-based person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13334–13343, 2021.

Yaoyao Liu, Bernt Schiele, and Qianru Sun. An ensemble of epoch-wise empirical bayes for few-shot learning. In *European Conference on Computer Vision*, pp. 404–421. Springer, 2020e.

Xu Luo, Longhui Wei, Liangjian Wen, Jinrong Yang, Lingxi Xie, Zenglin Xu, and Qi Tian. Rectifying the shortcut learning of background: Shared object concentration for few-shot image recognition. *arXiv preprint arXiv:2107.07746*, 2021.

Puneet Mangla, Nupur Kumari, Abhishek Sinha, Mayank Singh, Balaji Krishnamurthy, and Vineeth N Balasubramanian. Charting the right manifold: Manifold mixup for few-shot learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2218–2227, 2020.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. 2016.

Mamshad Nayeem Rizve, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Exploring complementary strengths of invariant and equivariant representations for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10836–10846, 2021.

Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *arXiv preprint arXiv:1807.05960*, 2018.

Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. *arXiv preprint arXiv:1804.10745*, 2018.

Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtash Harandi. Adaptive subspaces for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4136–4145, 2020.

Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017.

Hao Sun, Zhiqun Zhao, and Zhihai He. Reciprocal learning networks for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7416–7425, 2020.

Jiamei Sun, Sebastian Lapuschkin, Wojciech Samek, Yunqing Zhao, Ngai-Man Cheung, and Alexander Binder. Explanation-guided training for cross-domain few-shot classification. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 7609–7616. IEEE, 2021.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1199–1208, 2018.

Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pp. 266–282. Springer, 2020.

Hung-Yu Tseng, Hsin-Ying Lee, Jia-Bin Huang, and Ming-Hsuan Yang. Cross-domain few-shot classification via learned feature-wise transformation. *arXiv preprint arXiv:2001.08735*, 2020.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8769–8778, 2018.

Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pp. 6438–6447. PMLR, 2019.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29:3630–3638, 2016.

Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *arXiv preprint arXiv:1805.12018*, 2018.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

Haoqing Wang and Zhi-Hong Deng. Cross-domain few-shot classification via adversarial task augmentation. *arXiv preprint arXiv:2104.14385*, 2021.

Yan Wang, Wei-Lun Chao, Kilian Q Weinberger, and Laurens van der Maaten. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning. *arXiv preprint arXiv:1911.04623*, 2019.

Davis Wertheimer, Luming Tang, and Bharath Hariharan. Few-shot classification with feature map reconstruction networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8012–8021, 2021.

Chenchu Xu, Joanne Howey, Pavlo Ohorodnyk, Mike Roth, Heye Zhang, and Shuo Li. Segmentation and quantification of infarction without contrast agents via spatiotemporal generative adversarial learning. *Medical image analysis*, 59:101568, 2020.

Ling Yang, Liangliang Li, Zilun Zhang, Xinyu Zhou, Erjin Zhou, and Yu Liu. Dpgn: Distribution propagation graph network for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13390–13399, 2020a.

Xu Yang, Cheng Deng, Kun Wei, Junchi Yan, and Wei Liu. Adversarial learning for robust deep clustering. *Advances in Neural Information Processing Systems*, 33, 2020b.

Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8808–8817, 2020.

Baoquan Zhang, Xutao Li, Yunming Ye, Zhichao Huang, and Lisai Zhang. Prototype completion with primitive knowledge for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3754–3762, June 2021a.

Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12203–12213, 2020.

Lu Zhang, Shuigeng Zhou, Jihong Guan, and Ji Zhang. Accurate few-shot object detection with support-query mutual guidance and hybrid loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14424–14432, 2021b.

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

Imtiaz Ziko, Jose Dolz, Eric Granger, and Ismail Ben Ayed. Laplacian regularized few-shot learning. In *International Conference on Machine Learning*, pp. 11660–11670. PMLR, 2020.

A Appendix

In this appendix, we provide more details of experimental settings and additional results to further understand the performance of our proposed method.

A.1 Dataset

In our experiments, the following 5 datasets are used for performance evaluations and comparisons.

(1) Mini-ImageNet has randomly selected 100 categories from the ImageNet (Deng et al., 2009) and each category has 600 samples of size 84×84. The 100 categories are divided into a training set with 64 categories, a validation set with 16 categories, and a testing set with 20 categories. **(2) CUB** is a fine-grained dataset with 200 bird species mainly living in North America (Wah et al., 2011). We randomly split the dataset into 100, 50, 50 classes for training, validation and testing, respectively. **(3) Cars** contains 16,185 images of 207 fine-grained car types, which consist of 10 BMW models and 197 other car types (Krause et al., 2013). We randomly selected 196 categories include 98 training, 49 validation and 49 testing for the experiment. **(4) Places** is a dataset of scene images (Zhou et al., 2017), containing 73,000 training images from 365 scene categories, which are divided into 183 categories for training, 91 for validation and 91 for testing. **(5) Plantae** is a sub-set of the iNat2017 dataset (Van Horn et al., 2018), which contains 200 types of plants and a total of 47,242 images. We split them into 100 classes for training, 50 for validation, and 50 for testing.

The Mini-ImageNet is the most popular benchmark for few-shot classification. It is usually used as a baseline dataset for model training. The CUB dataset is more frequently used for few-shot fine-grained classification tasks. The Cars, Places and Plantae datasets are used for model testing in cross-domain few-shot classification tasks.

A.2 The visualization of feature in self-supervised Learning.

The proposed SPDN method incorporates the self-supervised constraint into the training process, aiming to improve the quality of learned features and the generalization capability of the few-shot learning. Figure. 6 shows the tSNE visualization of the learned features of 100 samples from the mini-ImageNet dataset for each class in a 5-way 5-shot setting. We can see that, with the self-supervised learning, the features of each class are more concentrated into clusters.

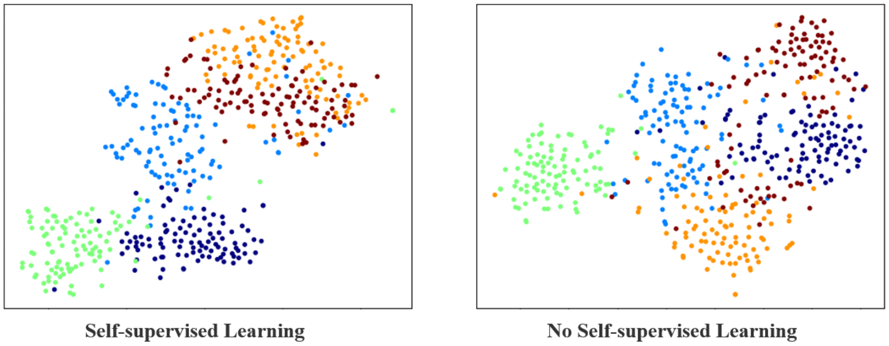


Figure 6: The tSNE visualization of feature with or without Self-supervised Learning.

A.3 self-supervised optimization (SSO) modules

The proposed self-supervised optimization (SSO) modules aim to correct the predicted query labels. In the following experiment, we are trying to understand how many incorrect label prediction of the query labels have been successfully corrected by the SSO module. Table 5 shows the results from the 5-way 1-shot on the CUB dataset. We keep track of 75 randomly selected query samples. If we predict the query labels only using the prime network without using the SSO (before SSO), the number of query samples with incorrect labels is 57, and the number of correct ones is 18, which is very low. After we apply the SSO, the number of query samples with incorrect labels is reduced to 45, the number of correct ones increases to 30. We can examine this correction process in more detail. The SSO module has corrected the labels for 15 samples, as shown in the third row (Incorrect → Correct Label) of the table. However, it has also mis-corrected the labels for 3 samples, as shown in the last row (Correct → Incorrect Label) of the table. In our experiments, we have observed that the SSO module is able to correct the labels for much more query samples than those mis-corrected one. This implies that the dual network and the self-supervision constraint are working very well for few-shot learning. This explains the significant performance achieved by the proposed self-supervised prime-dual network method.

Table 5: The predicted query labels before and after the SSO.

Before SSO	57	18
After SSO	45	30
Incorrect → Correct Label	15	
Incorrect → Incorrect Label	42	
Correct → Correct Label	15	
Correct → Incorrect Label	3	

A.4 Extension to N-Shot Image Classification

In the main paper, we have used the 5-way 1-shot image classification as an example to present our method of self-supervised prime-dual network (SPDN) and optimization for few-shot image classification. This method can be naturally extended to generic K-way N-shot image classification. Figure 7 illustrates an example of extension to 5-way 5-shot. In this case, each class, in both training and test stages, has 5 support samples and one query sample. In the prime network, we use these 5 support samples to predict the label of the query sample. To ensure that the dual network shares the same network structure as the prime network, for the reverse prediction, we randomly select one sample (denoted by $\mathbf{s_0}$) from the support set and switch it with the query sample $\mathbf{q}_0$. During the training and inference of the dual network, this updated support set is used to predict the label of $\mathbf{s_0}$, which is then compared to its ground-truth label to compute the self-supervised loss. This loss is used for joint prime-dual network training, as well as the self-supervised optimization of the label prediction for the query sample $\mathbf{q}_0$.

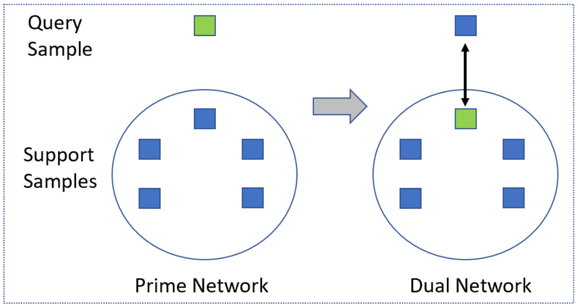


Figure 7: Illustration of extension to 5-shot classification.

A.5 Further Understanding of the Self-Supervised Optimization of Query Label Prediction

In our proposed SPDN method, the self-supervised optimization of the query label prediction plays an important role and improves the performance significantly. In this section, we provide more experimental results to demonstrate and further under the performance of this algorithm module. Figure 8 shows 6 examples of 5-way 1-shot image classification. Initially, the predicted label for these query samples are incorrect. Then, we perform self-supervised search of the query labels within the neighborhood of the predicted label. We use this predicted labels as input to the dual network to predict the labels of the support samples. The label prediction error of the support samples is used as the optimization objective. In Figure 8, under each query sample, we show the decreasing of the optimization objective (support label error) with the number of searched candidate query labels. These results show that it is sufficient to search 5-8 candidate query label vectors.

It should be noted that the self-supervised optimization query label prediction can correct the incorrect label prediction, adjusting incorrect label prediction into correct ones. Certainly, it will make mistake or mis-correct the query label prediction, adjusting correct label predictions into incorrect ones. However, the probability of the mis-correction is much lower. For example, Table 6 shows percentages of correct adjustment and incorrect adjustment by the optimization module on the Cars dataset. Specifically, the percentage of correct adjustment from incorrect query labels into correct ones is 21.6%. In the meantime, the percentage of incorrect adjustment is 5.7%. This result in a performance improvement of 15.8% in the overall few-shot image classification, from 32.8% to 48.6%, which is quite significant.

Table 6: The predicted query labels before and after the SSO.

	Incorrect	Correct
Before SSO	67.2%	32.8%
After SSO	51.3%	48.6%
Incorrect → Correct Label	21.6%	
Incorrect → Incorrect Label	45.6%	
Correct → Correct Label	27.1%	
Correct → Incorrect Label	5.7%	

A.6 Further Discussion of the Proposed Method

The key idea and motivation behind our dual network design is as follows: one central challenge in network prediction is that we have no ways to check if the prediction is accurate or not, since we do not have the ground truth. To address this issue, we develop the prime-dual network structure, where the successfully learned dual network is used as a verification module to verify if the prediction results are good enough or not. It maps the prediction results back to the current known data. We establish the self-supervised loss defined on the current known data, use it as the objective function to perform local search and refinement of the prediction results. This process is unique and contributes significantly to the overall performance. The prime network is the baseline GNN+FT network using support samples to predict query samples. The dual network is another GNN+FT network (in opposite direction) using query samples to predict support samples. These two networks form a prediction loop and a self-supervised loss is then derived. We implement this new idea on the the GNN+FT few-shot learning method to demonstrate its performance. The proposed idea is generic and can be applied to other methods, even in other prediction and learning problems, which will be studied in our future work. Our proposed idea is new. However, it does introduce additional complexity. According to our estimation, it will add about 40-60% extra complexity on top of the existing baseline since a majority of computation, such as feature extraction, does not need to re-computed during the search process. In our future work, we plan to develop schemes to reduce the complexity of the self-supervised optimization, for example by merging multiple search steps into one execution cycle.

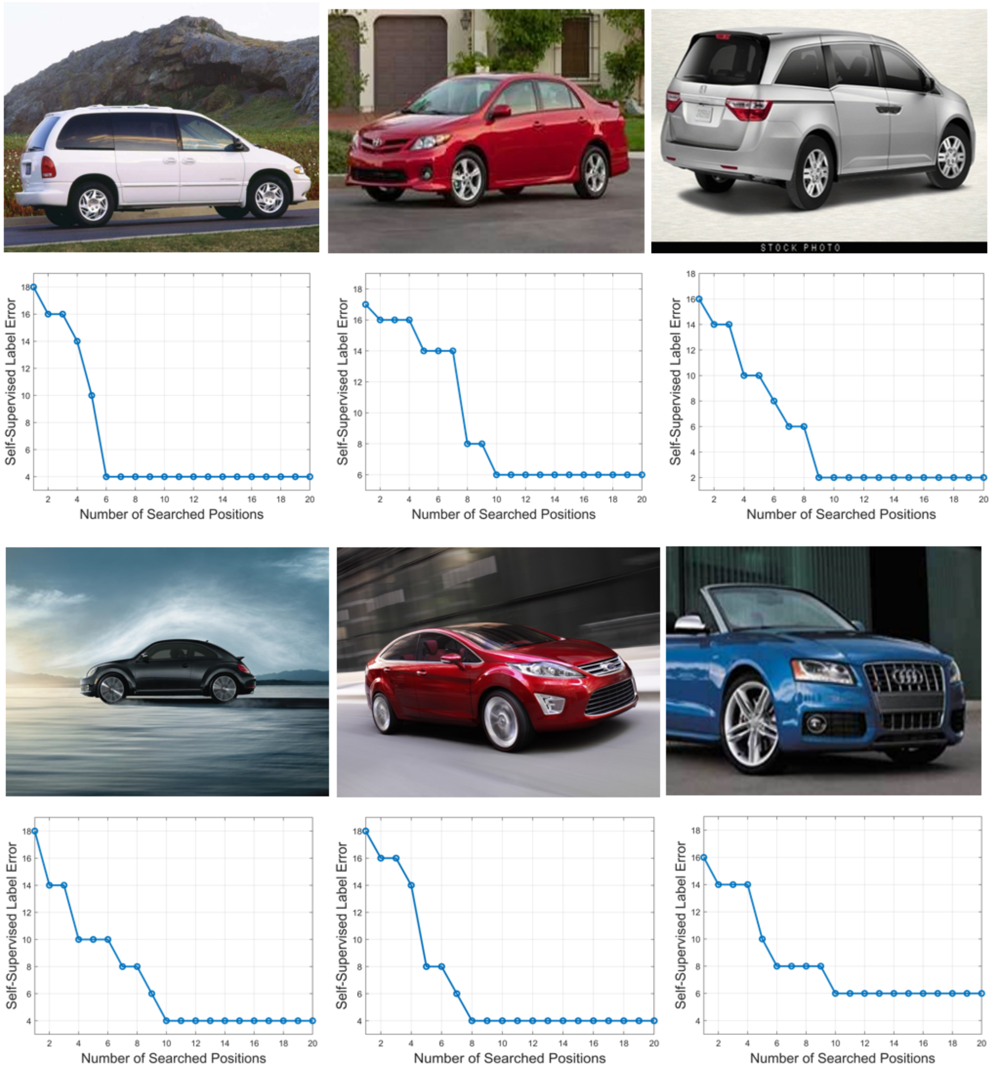


Figure 8: The decreasing of label error for self-supervised optimization of label prediction of query samples from the Cars dataset.