A Density-driven Iterative Prototype Optimization for Transductive Few-shot Learning

Jingcong Li, **Chunjin Ye**, **Fei Wang** and **Jiahui Pan***
School of Software, South China Normal University
lijingcong@hotmail.com, yechunjin@m.scnu.edu.cn, fwang@scnu.edu.cn, panjiahui@m.scnu.edu.cn

Abstract

Few-shot learning (FSL) poses a considerable challenge since it aims to improve the model generalization ability with limited labeled data. Previous works usually attempt to construct class-specific prototypes and then predict novel classes using these prototypes. However, the feature distribution represented by the limited labeled data is coarse-grained, leading to large information gap between the labeled and unlabeled data as well as biases in the prototypes. In this paper, we investigate the correlation between sample quality and density, and propose a Density-driven Iterative Prototype Optimization to acquire high-quality prototypes, and further improve few-shot learning performance. Specifically, the proposed method consists of two optimization strategies. The similarity-evaluating strategy is for capturing the information gap between the labeled and unlabeled data by reshaping the feature manifold for the novel feature distribution. The density-driven strategy is proposed to iteratively refine the prototypes in the direction of density growth. The proposed method could reach or even exceed the state-of-the-art performance on four benchmark datasets, including *mini*ImageNet, *tiered*ImageNet, CUB, and CIFAR-FS. The code will be available soon at https://github.com/tailofcat/DIPO.

1 Introduction

Recently, the significance of employing powerful feature embedding has been comprehensively investigated in FSL [Chen *et al.*, 2019; Dhillon *et al.*, 2019; Chen *et al.*, 2020; Tian *et al.*, 2020]. The initial step in most typical FSL methods is to pre-train a feature extractor using the base classes. Then, for each meta-testing task, the features extracted from all images in the support set are utilized to construct a task-specific classifier, facilitating the recognition of image categories within the query set. Although a powerful feature extractor can acquire a suitable feature space for clustering unseen data, the domain shift between training and testing [Lichtenstein *et al.*, 2020; Tao *et al.*, 2022], along with the information gap between support and query [Liu *et al.*, 2022; Dang *et al.*, 2023], pose a formidable challenge for estimating novel class distributions using limited support data. Consequently, the classification performance is directly influenced by the sample quality of the class prototype. If the prototype lacks adequate recognizable information to represent the entire class and contains excessive intra-class diversity to be outlier data, then the constructed classifier may exhibit bias.

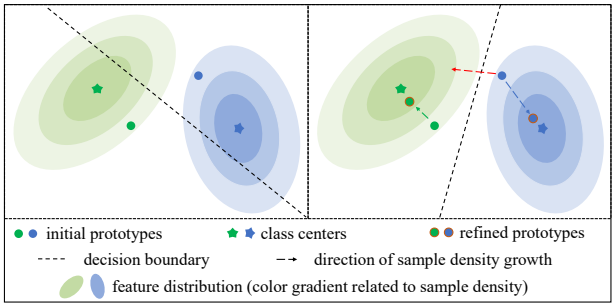

Figure 1: Illustration of the feature distributions in the feature space. The decision boundary is highly sensitive to the quality of the class-specific prototypes, and it may be confused by the prototypes in the lower-density regions (left). Refining the prototypes in the right direction (green and blue arrows) of sample density growth improves the decision boundary (right).

In our work, we investigate the correlation between sample quality and density, revealing that sample density plays a crucial role. Intuitively speaking, the direction of sample density growth can be considered the direction in which recognizable information aggregates. In higher-density regions, samples are tightly clustered within a class, which indicates the wealth of recognizable information (or the representative pattern) contained in those regions. Conversely, samples are sparsely observed in lower-density regions. The increased intra-class distance in those regions suggests that the samples tend to be more diverse and have less recognizable information. In other words, the class-specific prototypes in lower-density regions have a lower quality that may confuse the classifier, as shown in Figure 1(left). Thus, to enhance the

*Corresponding author
The supplementary material is available at: https://zenodo.org/records/11102557.

quality of the prototype, refining it in the direction of sample density growth is necessary. However, only knowing limited support data is inadequate for obtaining a reliable approximation, the estimation of sample density within a class is biased. To minimize the bias of empirical estimation, introducing query samples is the most straightforward way to expand the support set. Unfortunately, this may suffer because of the potentially large information gap between support and query, i.e., the non-estimable intra-class diversity of the novel class, leading to the wrong direction of sample density growth within a class, as shown in Figure 1(right).

Here, we propose a similarity-evaluating strategy to evaluate the information gap between support and query. The similarity-evaluating strategy seeks the base classes for assistance based on the insight: in the feature space, samples from a novel class tend to cluster in a certain direction [Tao *et al.*, 2022], which may indicate that the novel class skewed to that direction has a strong correlation with these base classes. Besides, the covariance matrix contains the underlying distribution information that can effectively assess the intra-class diversity of samples from the same feature distribution [Li *et al.*, 2019; Gao *et al.*, 2018]. Thus, we estimate the covariance matrix of the novel class using the most similar k base classes to the support data (with the closest angular similarity). By normalizing the feature distribution of the novel class using the covariance matrix, the similarity-evaluating strategy can reshape the feature manifold, thus effectively evaluating the information gap between support and query. In addition, we propose a density-driven strategy to improve the quality of the prototype in the direction of density growth. As the prototype progressively gets closer to the region with the local density maximum, the intra-class diversity of the class prototype declines and its representativeness rises.

The similarity-evaluating strategy and the density-driven strategy constitute the Density-driven Iterative Prototype Optimization (DIPO) that we propose in this work. The DIPO is generic, flexible, and can be applied agnostic to the pre-trained feature extractor without any extra learnable parameters. By evaluating the information gap between labeled and unlabeled samples and refining the quality of class prototypes, DIPO can effectively enhance the decision boundary of the nearest neighbor classifier. The main contributions of our paper are as follows:

1. We introduce a novel prototype rectifying method by refining the quality of the prototype in the direction of sample density growth for FSL problem.

2. A similarity-evaluating strategy is proposed to adeptly capture the intra-class diversity to evaluate the information gap between labeled and unlabeled samples. Furthermore, a density-driven strategy is proposed to iteratively refine the prototype in the direction of density growth.

3. The proposed method could reach or even exceed the state-of-the-art performance on four benchmark datasets, including *mini*ImageNet, *tiered*ImageNet, CUB, and CIFAR-FS.

2 Related Work

Recently, many FSL techniques can be broadly categorized into three groups: optimization-based methods, metric-based methods, and generation-based methods. The optimization-based approaches, such as MAML [Finn *et al.*, 2017] and LEO [Rusu *et al.*, 2019], concentrate on learning a set of model parameters that allow for fast gradient updates of the initial model for quick adaptation to new tasks. The metric-based methods aim to encode data in a suitable feature embedding space and predict them using distance-based classifiers, such as MatchingNet [Vinyals *et al.*, 2016], ProtoNet [Snell *et al.*, 2017], RelationNet [Sung *et al.*, 2018], etc. The generation-based methods aim to address data scarcity by learning data generation strategies, including AFHN [Li *et al.*, 2020], Matchinggan [Hong *et al.*, 2020], and DC [Yang *et al.*, 2021]. More recently, some works [Chen *et al.*, 2019; Dhillon *et al.*, 2019; Chen *et al.*, 2020; Tian *et al.*, 2020] reveal the significance of employing powerful feature embedding in FSL. This paradigm typically trains a general feature extractor on base classes and then learns a class-specific classifier with the novel classes.

More recently, some studies have explored transductive inference in FSL. Unlike the inductive setting, which can only utilize labeled samples, the transductive setting can access all the query data to restrict the hypothesis space for the novel classes. Based on this observation, some methods have attempted to learn high-quality prototypes in a transductive setting. BD-CSPN [Liu *et al.*, 2020b] is proposed for prototype rectification by label propagation and feature shifting. Lichtenstein [Lichtenstein *et al.*, 2020] aims to utilize subspace learning to extract discriminant features and then perform classification with the prototypes. PT+MAP [Hu *et al.*, 2021] utilizes an optimal-transport-inspired algorithm to adjust the class prototypes. MetaNODE [Zhang *et al.*, 2022] employs a meta-learning-based optimization framework to rectify prototypes. protoLP [Zhu and Koniusz, 2023] improves prototype estimation through prototype-based label propagation. Although these methods exhibit effectiveness to some extent, they are also affected by the information gap [Liu *et al.*, 2022] between labeled and unlabeled samples, leading to bias in prototype estimation. Different from them, our method aims to diminish the damage caused by the potential information gap and proposes a novel perspective for prototype estimation in FSL by refining the quality of the prototype in the direction of sample density growth.

3 Method

3.1 Problem Definition

We follow a common few-shot classification framework. Given a dataset with data-label pairs $\{(x_i, y_i)\}$ where $x_i \in \mathbb{R}^d$ is the d-dimensional feature vector of a sample and $y_i \in C$ is the corresponding label. C represents the set of all base classes C_b, validation classes C_v, and novel classes C_n, where $C_b \cap C_v \cap C_n = \emptyset$ and $C_b \cup C_v \cup C_n = C$. For the N-way K-shot task $\mathcal{T}$, there are a support set $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{N \times K}$ and a query set $\mathcal{Q} = \{(x_i, y_i)\}_{i=1}^{N \times q}$, where N is the number of classes within the task, and K and q are the labeled support samples and unlabeled query samples per

novel class, respectively. The novel class dataset D_{novel} is defined as $D_{novel} = \mathcal{S} \cup \mathcal{Q}$. Furthermore, there is a fully annotated dataset comprising abundant labeled images from base classes, designated as $D_{base} = \{(x_i, y_i)\}_{i=1}^{B}$, where B denotes the total number of images within D_{base}.

3.2 Similarity-Evaluating Strategy

Many transductive FSL methods focus on leveraging the latent relationships between support and query samples, aiming to maximize the utility of few-shot datasets. However, given that a few labeled samples can solely represent the local feature distribution (especially in a 1-shot setting), the intra-class diversity of the novel class is hard to estimate. Consequently, these methods suffer from the information gap between support and query, i.e., the non-estimable intra-class diversity of the novel class, which makes it difficult to model the latent relationships between labeled and unlabeled samples. We propose an easy-plug-in similarity-evaluating strategy to effectively capture the information gap between support and query.

Step 1: Seeking the relevant base classes. The similarity-evaluating strategy is based on the insight that the domain shift between the base and novel classes results in the skewness of the novel class distribution in a certain direction [Tao *et al.*, 2022].This may indicate that the novel class skewed in that direction has a strong correlation with these base classes. Inspired by the insight, we employ a mechanism to transfer the diversity of the relevant base classes, thereby quantifying the non-estimable intra-class diversity inherent in the novel class. Specifically, we first calculate the mean feature of each novel class using the following formulation:

$$\mu_j = \frac{1}{|\mathcal{S}_j|} \sum_{(x_i, y_i) \in \mathcal{S}_j} x_i, \tag{1}$$

where $\mathcal{S}_j$ denotes the subset of samples belonging to the j-th novel class within the support set, expressed as $\mathcal{S} = \{\mathcal{S}_1, ..., \mathcal{S}_N\}$. The notation $|\mathcal{S}_j|$ signifies the count of samples contained within $\mathcal{S}_j$, x_i represents the feature vector corresponding to the sample i, and y_i denotes the assigned label.

We then select the top k base classes based on the closest angular similarity between the mean of the support features and the mean of the features from base classes, as quantified by the cosine distance:

$$\begin{aligned} \mathbb{D}_j &= \{cos(\mu_j, \mu_i^b) \mid i \in C_b\}, \\ \mathbb{I}_j &= \{i \mid cos(\mu_j, \mu_i^b) \in Top_k(\mathbb{D}_j)\}, \end{aligned} \tag{2}$$

where μ_i^b is the mean of the features from base class i, the notation $cos(\cdot, \cdot)$ represents the cosine distance, and $Top_k(\cdot)$ represents the operator for selecting k elements possessing the highest values from $\mathbb{D}_j$. $\mathbb{I}_j$ contains the most relevant k base classes with respect to the mean feature μ_j.

Step 2: Making the distribution more Gaussian-like. We assume that the features of each class follow a multivariate Gaussian distribution, where the intra-class diversity can be effectively embedded in a covariance matrix. To reduce the domain shift as mentioned above and improve the consistency of the feature distribution with the Gaussian distribution, we transform the features in base and novel classes using Tukey's Ladder of Power transform [Tukey, 1977]:

$$\hat{x}_i = \begin{cases} x_i^{\beta}, & \text{if } \beta \neq 0 \\ \log(x_i), & \text{if } \beta = 0 \end{cases}, \tag{3}$$

where β denotes a hyperparameter utilized to fine-tune the skewness of the feature distribution. Modifying the value of β brings the distribution closer to a Gaussian-like distribution and makes the intra-class diversity across different classes more relevant. After the Tukey's transformation, the support set and the query set in the novel classes are updated to $\mathcal{S} = \{(\hat{x}_i, y_i)\}_{i=1}^{N \times K}$ and $\mathcal{Q} = \{(\hat{x}_i, y_i)\}_{i=1}^{N \times q}$.

Step 3: Normalizing the feature distribution for the novel class. We evaluate the information gap between support and query by normalizing the feature distribution for each novel class. More precisely, we first approximate the covariance matrix for each novel class using the weighted covariance of the relevant base classes. The weight factor is expressed as follows:

$$w_i = \frac{cos(\mu_j, \mu_i^b)}{\sum_{k \in \mathbb{I}_j} cos(\mu_j, \mu_k^b)}, \tag{4}$$

The approximated covariance of the novel class j is formulated as:

$$\begin{aligned} \Sigma_j &= \sum_{i \in \mathbb{I}_j} w_i \Sigma_i^b, \\ \Sigma_j &= \Sigma_j + \bar{\sigma} I, \end{aligned} \tag{5}$$

where Σ_i^b is the corresponding covariance matrix of the base class i after Tukey's transformation, and $\bar{\sigma}$ is the average diagonal variance of Σ_j, as the regularization term to prevent Σ_j from being non-invertible. By this approximation, the covariance of the novel class j is more accurate and stable than the covariance calculated using the support set. With only a few labeled samples (especially in a 1-shot setting), the covariance of the support set is unable to capture the entire intra-class diversity of the novel class. Given the correlation between the novel class and the relevant base classes,

Algorithm 1 The prediction procedure of the proposed DIPO for an N-way-K-shot task

Require: $\mathcal{D}_{base}$
Require: $\mathcal{D}_{novel} = \mathcal{S} \cup \mathcal{Q}$
1: Select the most similar base classes for each novel class (Equation 1, 2).
2: Transform the features from $\mathcal{D}_{base}$ and $\mathcal{D}_{novel}$ with Tukey's Ladder of Power Transform (Equation 3).
3: Approximate the covariance matrix for each novel class (Equation 4, 5).
4: Calculate the initial prototype for each novel class (Equation 7).
5: **for** $iter$ in $\{1, \cdots, N_{iter}\}$ **do**
6: Select the most similar query neighbors to form the neighborhood set for each novel class (Equation 8, 9).
7: Refine the initial prototype in the global direction of sample density growth for each novel class (Equation 10, 11).
8: **end for**
9: Predict the labels of query samples via a nearest neighbor classifier (Equation 12).

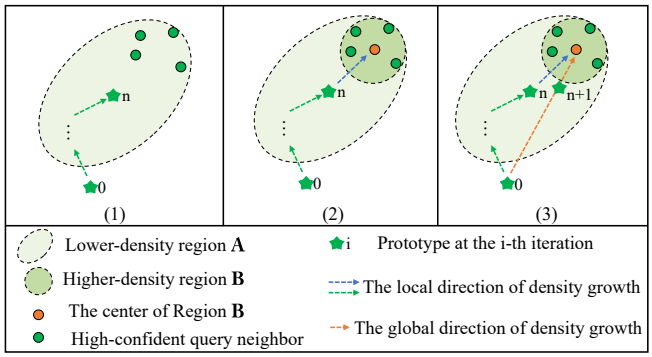

Figure 2: The process of the density-driven strategy for refining the initial prototype at the n-th iteration. Given the prototype of the novel class, the density-driven strategy consists of three steps: 1) identifying the lower-density region $\boldsymbol{A}$ using the high-confident query neighbors; 2) determining the local direction of sample density growth (the blue arrow) by locating the center of higher-density region $\boldsymbol{B}$; 3) refining the initial prototype in the global direction of sample density growth (the orange arrow).

extrapolating the intra-class diversity of the novel class from the base classes is considered a more accurate alternative.

For a feature vector within a class, there are locations encoding class-specific information and locations encoding class-independent information, thus reflecting the intra-class diversity. Values at class-specific locations tend to demonstrate high correlations, playing a significant role in distinguishing the class. Conversely, values derived from class-independent locations are similar among all samples, which indicates redundant information and has little effect on classification. To emphasize the locations encoding class-specific information and mitigate interference from the locations encoding class-independent information, we normalize the feature distribution for the novel class j as follows:

$$d(\hat{x}_i, \hat{x}_j) = \sqrt{(\hat{x}_i - \hat{x}_j)^T {\Sigma_j}^{-1} (\hat{x}_i - \hat{x}_j)}, \tag{6}$$

where $\hat{x}_i$ is the unlabeled feature after Tukey's transformation corresponding to the sample i in the query set, $\hat{x}_j$ is the labeled feature after Tukey's transformation corresponding to the sample j in $\mathcal{S}_j$, and Σ_j is the approximated covariance matrix of the novel class j. And the information gap across features under the same distribution is represented by the $d(\cdot,\cdot)$, which is known as the Mahalanobis distance. Through this normalization process, the feature manifold for the novel feature distribution is reshaped to amplify the distinctions between features. This effectively evaluates the information gap between features, thus facilitating the modeling of latent relationships between support and query.

3.3 Density-Driven Strategy

The sample quality of the class-specific prototype has a direct impact on the classification performance. We propose a density-driven strategy to improve the quality of the prototype. The density-driven strategy is based on the insight: in the feature space, there are higher-density regions and lower-density regions. Samples from higher-density regions are tightly clustered within a class to distinguish them well. Samples are sparsely observed in lower-density regions, which indicates they have a lower quality to separate well between classes. Therefore, the direction of sample density growth can be seen as the direction in which recognizable information aggregates. Consequently, we enhance the sample quality of the prototype in the direction of density growth.

The overview of the proposed density-driven strategy is shown in Figure 2, which consists of three steps at the n-th iteration: 1) Given the prototype of the novel class, we identify the lower-density region $\boldsymbol{A}$ for the novel class using the high-confident query neighbors from the query set. 2) We determine the local direction of sample density growth by locating the center of higher-density region $\boldsymbol{B}$. 3) We refine the initial prototype by the global direction of sample density growth. As the initial prototype progressively gets closer to the higher-density regions, the intra-class diversity of the prototype declines and its representativeness rises. Specifically, the initial class prototype for the novel class j is computed as the mean of support features after Tukey's transformation:

$$c_j^{(0)} = \frac{1}{|\mathcal{S}_j|} \sum_{(\hat{x}_i, y_i) \in \mathcal{S}_j} \hat{x}_i, \tag{7}$$

Unfortunately, with only a few samples, the prototype based on the mean of the support features is prone to overfitting on the support data and far from representing the whole novel class. Subsequently, the prototype is iteratively re-estimated in the manner described below:

Step 1: Seeking the query neighbors to identify the lower-density region. To estimate the sample density for novel class j around the prototype, we select some high-confident pseudo-labeled neighbors from the query set. Specifically, the confidence in classifying the query sample $\hat{x}_i$ into the novel class j at the n-th iteration is calculated as follows:

$$M_{ij} = \frac{\exp(-d(\hat{x}_i, c_j^{(n)}))}{\Sigma_{k=1}^{N} \exp(-d(\hat{x}_i, c_k^{(n)}))}, \tag{8}$$

where $c_j^{(n)}$ is the prototype of novel class j at the n-th iteration, and the $d(\cdot,\cdot)$ is the information gap between the query sample $\hat{x}_i$ and the prototype as formulated in Equation 6 in the similarity-evaluating strategy. As the similarity-evaluating strategy adeptly captures the information gap between features, it enables the generation of pseudo-labels with high quality. Subsequently, we select the top $m^{(n)}$ pseudo-labeled query samples with the highest confidence for each novel class, thereby constituting the neighborhood set as follows:

$$\begin{aligned} \mathbb{A}_j &= \{i \mid M_{ij} \in Top_m(M_{*j})\}, \\ \mathbb{N}_j &= \{\hat{x}_i \mid i \in \mathbb{A}_j, \hat{x}_i \in \mathcal{Q}\}, \end{aligned} \tag{9}$$

where $Top_m(\cdot)$ represents the operator for selecting $m^{(n)}$ elements with the highest confidence from each column of the matrix, and the hyperparameter $m^{(n)}$ denotes the total number of samples within the neighborhood set per novel class at the n-th iteration. $\mathcal{Q}$ denotes the unlabeled query set after Tukey's transformation. $\mathbb{A}_j$ contains the indices of the

			*mini*ImageNet		*tiered*ImageNet	
Method	Transductive	Backbone	1-shot	5-shot	1-shot	5-shot
MAML [Finn *et al.*, 2017]	✗	ResNet-18	49.61 ± 0.92	65.72 ± 0.77	–	–
RelationNet [Sung *et al.*, 2018]	✗	ResNet-18	52.48 ± 0.86	69.83 ± 0.68	–	–
MatchingNet [Vinyals *et al.*, 2016]	✗	ResNet-18	52.91 ± 0.88	68.88 ± 0.69	–	–
ProtoNet [Snell *et al.*, 2017]	✗	ResNet-18	54.16 ± 0.82	73.68 ± 0.65	–	–
TEAM [Qiao *et al.*, 2019]	✓	ResNet-18	60.07	75.9	–	–
Baseline [Chen *et al.*, 2019]†	✗	ResNet-18	50.28 ± 0.18	74.74 ± 0.15	–	–
Baseline + DIPO(ours)	✓	ResNet-18	$\mathbf{71.85 \pm 0.29}$	$\mathbf{83.50 \pm 0.16}$	–	–
DPGN [Yang *et al.*, 2020]	✓	ResNet-12	67.77 ± 0.32	84.60 ± 0.43	72.45 ± 0.51	87.24 ± 0.39
CAN+T [Hou *et al.*, 2019]	✓	ResNet-12	67.19 ± 0.55	80.64 ± 0.35	73.21 ± 0.58	84.93 ± 0.38
MetaNODE [Zhang *et al.*, 2022]	✓	ResNet-12	77.92 ± 0.99	85.13 ± 0.96	83.46 ± 0.92	88.46 ± 0.57
Inv-Equ [Rizve *et al.*, 2021]†	✗	ResNet-12	67.37 ± 0.19	84.30 ± 0.13	72.07 ± 0.22	86.50 ± 0.15
Inv-Equ + DIPO(ours)	✓	ResNet-12	$\mathbf{82.97 \pm 0.25}$	$\mathbf{90.13 \pm 0.12}$	$\mathbf{84.90 \pm 0.25}$	$\mathbf{90.30 \pm 0.14}$
Transductive Fine-tuning [Dhillon *et al.*, 2019]	✓	WRN-28-10	65.73 ± 0.68	78.40 ± 0.52	73.34 ± 0.71	85.50 ± 0.50
LaplacianShot [Ziko *et al.*, 2020]	✓	WRN-28-10	74.86 ± 0.19	84.13 ± 0.14	80.18 ± 0.21	87.56 ± 0.15
BD-CSPN [Liu *et al.*, 2020b]	✓	WRN-28-10	70.31 ± 0.93	81.89 ± 0.60	78.74 ± 0.95	86.92 ± 0.63
SIB [Hu *et al.*, 2020]	✓	WRN-28-10	70.00 ± 0.60	79.20 ± 0.40	–	–
Oblique Manifold [Qi *et al.*, 2021]	✓	WRN-28-10	80.64 ± 0.34	89.39 ± 0.39	85.22 ± 0.34	91.35 ± 0.42
iLPC [Lazarou *et al.*, 2021]	✓	WRN-28-10	83.05 ± 0.79	88.82 ± 0.42	88.50 ± 0.75	92.46 ± 0.42
noHub-S [Trosten *et al.*, 2023]	✓	WRN-28-10	82.0 ± 0.26	88.03 ± 0.13	82.85 ± 0.27	88.31 ± 0.16
S2M2-R [Mangla *et al.*, 2020]†	✗	WRN-28-10	65.15 ± 0.20	83.20 ± 0.13	72.89 ± 0.22	88.00 ± 0.14
S2M2-R + PT+MAP [Hu *et al.*, 2021]†	✓	WRN-28-10	82.62 ± 0.24	88.82 ± 0.13	88.16 ± 0.22	92.14 ± 0.13
S2M2-R + EASE [Zhu and Koniusz, 2022] †	✓	WRN-28-10	82.97 ± 0.25	88.81 ± 0.13	88.60 ± 0.23	92.28 ± 0.13
S2M2-R + protoLP [Zhu and Koniusz, 2023]†	✓	WRN-28-10	82.92 ± 0.26	88.79 ± 0.13	88.50 ± 0.24	92.27 ± 0.14
S2M2-R + DIPO(ours)	✓	WRN-28-10	$\mathbf{84.41 \pm 0.24}$	$\mathbf{90.70 \pm 0.11}$	$\mathbf{88.81 \pm 0.23}$	$\mathbf{93.07 \pm 0.12}$

Table 1: The 5-way, 1-shot and 5-shot classification accuracy (%) on *mini*ImageNet and *tiered*ImageNet with 95% confidence intervals. "†" indicates that we have re-implemented the method in the same few-shot setting. The best results are highlighted in bold.

top $m^{(n)}$ high-confident query neighbors with respect to the prototype $c_j^{(n)}$. $\mathbb{N}_j$ is the neighborhood set of novel class j, which comprises the query samples corresponding to the indices stored in $\mathbb{A}_j$. The neighborhood set represents the local neighborhood information of the prototype, which can be used to describe the density. Thus, the lower-density region $\boldsymbol{A}$ for novel class j is delineated by the prototype $c_j^{(n)}$ and the query neighbors within the neighborhood set $\mathbb{N}_j$, as shown in Figure 2(1).

Step 2: Determining the local direction of sample density growth. Inspired by the mean shift [Fukunaga and Hostetler, 1975], we determine the local direction of sample density growth by locating the center of the higher-density region. More precisely, the center of the higher-density region $\boldsymbol{B}$ for novel class j at the n-th iteration is calculated as:

$$C_{\boldsymbol{B}}^{(n)} = \frac{1}{m^{(n)}} \sum_{\hat{x}_i \in \mathbb{N}_j} \hat{x}_i, \tag{10}$$

where $\mathbb{N}_j$ is the neighborhood set of novel class j, and $m^{(n)}$ is the number of samples within the neighborhood set $\mathbb{N}_j$. The higher-density region $\boldsymbol{B}$ for novel class j is located by the center $C_{\boldsymbol{B}}^{(n)}$ and the query neighbors within the neighborhood set $\mathbb{N}_j$, as shown in Figure 2(2).

The direction from the center of the lower-density region to the center of the higher-density region is the mean shift vector, which always points towards the direction of the maximum increase in the density [Fukunaga and Hostetler, 1975]. Consequently, the direction from the prototype $c_j^{(n)}$ to the center $C_{\boldsymbol{B}}^{(n)}$ can be interpreted as the local direction of density growth.

Step 3: Refining the initial prototype in the global direction of density growth. For a more stable and accurate estimation, we refine the prototype by taking into account all local directions of sample density growth from previous iterations. Specifically, the refined prototype is formulated as:

$$c_j^{(n+1)} = (1-\lambda) c_j^{(0)} + \lambda C_{\boldsymbol{B}}^{(n)}, \tag{11}$$

where $c_j^{(0)}$ is the initial prototype of novel class j, $C_{\boldsymbol{B}}^{(n)}$ is the center of the higher-density region $\boldsymbol{B}$ for novel class j at the n-th iteration, and $\lambda = \frac{m^{(n)}}{|\mathcal{S}_j| + m^{(n)}}$. This indicates that as the hyperparameter $m^{(n)}$ increases, the refined prototype gradually moves closer to $C_{\boldsymbol{B}}^{(n)}$ from $c_j^{(0)}$, which can be interpreted as the global direction of sample density growth, as shown in Figure 2(3). To avoid making risky decisions in the early iterations and to obtain a more stable and accurate prototype, we propose to slowly increase the number of query neighbors $m^{(n)}$ during the iterations. For simplicity, we define a tuple $(m, N_{iter}, step)$ to illustrate $m^{(n)}$ at each iteration, where m indicates the number of query neighbors for the first iteration, N_{iter} denotes the number of iterations, and $step$ represents the incremental increase in m for each subsequent iteration in the density-driven strategy. Therefore, the refined prototype $c_j^{(N_{iter})}$ can be regarded as the final prototype of novel class j.

3.4 Final Classification

We employ a nearest neighbor classifier as the task-specific classifier for each meta-test task. Specifically, the confidence of the query sample $\hat{x}_i$ belonging to class j is estimated based on the information gap between it and the prototype as follows:

$$P(y_i = j | \hat{x}_i) = \frac{\exp(-d(\hat{x}_i, c_j^{(N_{iter})}))}{\Sigma_{n=1}^{N} \exp(-d(\hat{x}_i, c_n^{(N_{iter})}))}, \tag{12}$$

where $c_j^{(N_{iter})}$ is the final prototype of class j obtained through the density-driven strategy, and the information

			CUB		CIFAR-FS	
Method	Transductive	Backbone	1-shot	5-shot	1-shot	5-shot
MAML [Finn *et al.*, 2017]	✗	ResNet-18	69.96 ± 1.01	82.70 ± 0.65	–	–
RelationNet [Sung *et al.*, 2018]	✗	ResNet-18	72.36 ± 0.90	83.64 ± 0.60	–	–
MatchingNet [Vinyals *et al.*, 2016]	✗	ResNet-18	67.59 ± 1.02	82.75 ± 0.58	–	–
ProtoNet [Snell *et al.*, 2017]	✗	ResNet-18	71.88 ± 0.91	87.42 ± 0.48	–	–
Negative-Cosine [Liu *et al.*, 2020a]	✗	ResNet-18	72.66 ± 0.85	89.40 ± 0.43	–	–
LaplacianShot [Ziko *et al.*, 2020]	✓	ResNet-18	80.96	88.68	–	–
Baseline [Chen *et al.*, 2019]†	✗	ResNet-18	59.96 ± 0.21	81.95 ± 0.13	–	–
Baseline + DIPO(ours)	✓	ResNet-18	**85.37 ± 0.24**	**91.30 ± 0.10**	–	–
DPGN [Yang *et al.*, 2020]	✓	ResNet-12	75.71 ± 0.47	91.48 ± 0.33	77.90 ± 0.50	90.20 ± 0.40
ECKPN [Chen *et al.*, 2021]	✓	ResNet-12	**77.43 ± 0.54**	**92.21 ± 0.41**	79.20 ± 0.40	91.00 ± 0.50
Inv-Equ [Rizve *et al.*, 2021]†	✗	ResNet-12	–	–	77.69 ± 0.21	89.44 ± 0.14
Inv-Equ + DIPO(ours)	✓	ResNet-12	–	–	**88.37 ± 0.22**	**91.94 ± 0.15**
Transductive Fine-tuning [Dhillon *et al.*, 2019]	✓	WRN-28-10	–	–	76.58 ± 0.68	85.79 ± 0.50
BD-CSPN [Liu *et al.*, 2020b]	✓	WRN-28-10	87.45	91.74	–	–
iLPC [Lazarou *et al.*, 2021]	✓	WRN-28-10	91.03 ± 0.63	94.11 ± 0.30	86.51 ± 0.75	90.60 ± 0.48
SIB [Hu *et al.*, 2020]	✓	WRN-28-10	–	–	80.00 ± 0.60	85.30 ± 0.40
S2M2-R [Mangla *et al.*, 2020]†	✗	WRN-28-10	80.32 ± 0.20	90.86 ± 0.11	74.66 ± 0.21	87.65 ± 0.15
S2M2-R + PT+MAP [Hu *et al.*, 2021]†	✓	WRN-28-10	91.56 ± 0.18	94.09 ± 0.09	87.36 ± 0.23	90.71 ± 0.16
S2M2-R + protoLP [Zhu and Koniusz, 2023]†	✓	WRN-28-10	91.67 ± 0.19	94.07 ± 0.10	**87.46 ± 0.24**	90.53 ± 0.16
S2M2-R + DIPO(ours)	✓	WRN-28-10	**92.52 ± 0.18**	**94.97 ± 0.08**	86.66 ± 0.23	**90.77 ± 0.15**

Table 2: The 5-way, 1-shot and 5-shot classification accuracy (%) on CUB and CIFAR-FS with 95% confidence intervals. "†" indicates that we have re-implemented the method in the same few-shot setting. The best results are highlighted in bold.

gap $d(\cdot,\cdot)$ is evaluated using Equation 6 in the similarity-evaluating strategy. Algorithm 1 shows the prediction procedure of the proposed DIPO.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate our approach on four representative few-shot classification benchmarks, including *mini*ImageNet, *tiered*ImageNet, CIFAR-FS, and CUB. ***mini*ImageNet** constitutes a subset of ImageNet [Russakovsky *et al.*, 2014]. Following the previous work [Ravi and Larochelle, 2017], the dataset is categorized into 64 base classes, 16 validation classes, and 20 novel classes. ***tiered*ImageNet** represents a more extensive subset of ImageNet [Russakovsky *et al.*, 2014], featuring 608 classes derived from a hierarchical category structure. Consistent with the previous work [Ren *et al.*, 2018], the dataset is partitioned into 351 base classes, 97 validation classes, and 160 novel classes. **CUB** is a fine-grained dataset that encompasses 200 distinct bird classes. Following [Welinder *et al.*, 2010], we divide the dataset into 100 base classes, 50 validation classes, and 50 novel classes. Following [Bertinetto *et al.*, 2018], **CIFAR-FS** is split into 64 base classes, 16 validation classes, and 20 novel classes.

Implementation Details. For the feature extractor, we employ the ResNet-18 [Chen *et al.*, 2019], ResNet-12 [Rizve *et al.*, 2021] and WRN-28-10 [Mangla *et al.*, 2020] to show the effectiveness of our method. Note that the features are extracted from the penultimate layer (with a ReLU activation function) of the feature extractor, ensuring that all values are non-negative to maintain the validity of Tukey's transformation in Equation 3. For the task-specific classifier, we employ a nearest neighbor classifier to establish the effective decision boundaries for the novel classes. The hyperparameters of the proposed method contain the number of relevant base classes k, the power of Tukey's transformation β, and the tuple $(m, N_{iter}, step)$, where m is the number of query neighbors for each novel class at the first iteration, N_{iter} is the number of iterations, and $step$ is the incremental increase in m for each subsequent iteration. More detailed information on the hyperparameter settings and the hyperparameter tuning is presented in the supplementary material. We follow the 5-way 1-shot and 5-way 5-shot settings and use the top-1 accuracy as the evaluation metric to evaluate the performance of our method. Unless otherwise specified, there are 15 query samples per class in each task. The reported results represent the average accuracy (%) along with the 95% confidence interval, computed over 10,000 tasks randomly selected from the novel classes.

4.2 Experimental Results

Few-shot image classification. We compare the proposed DIPO with state-of-the-art methods in the 5-way 1-shot and 5-way 5-shot settings. Table 1 shows the results on *mini*ImageNet and *tiered*ImageNet, and Table 2 shows the results on CUB and CIFAR-FS. Our DIPO is insensitive to feature extractors, and we employ the algorithms proposed in Baseline [Chen *et al.*, 2019], Inv-Equ [Rizve *et al.*, 2021], and S2M2-R[Mangla *et al.*, 2020] to obtain the pre-trained feature extractor to show the effectiveness of our method. Compared to the baseline models [Chen *et al.*, 2019; Rizve *et al.*, 2021; Mangla *et al.*, 2020], our method exhibits an average improvement of nearly 16% in the 1-shot task and nearly 6% in the 5-shot task. Among the state-of-the-art methods, BD-CSPN, SIB, MetaNODE, PT+MAP, and protoLP are our strong competitors since they also focus on learning high-quality prototypes for the novel classes. Compared to the competitors, we can observe that the DIPO achieves consistent improvements on different datasets. And the improvements are larger in the 1-shot task than the 5-shot task since the prototype is more biased in the 1-shot setting. This further illustrates the effectiveness of our approach in refining the prototypes. From these experiments, we can conclude that our proposed DIPO could reach or exceed the state-of-the-art methods under the 5-way 1-shot and 5-way 5-shot settings of *mini*ImageNet, *tiered*ImageNet, CIFAR-FS, and CUB.

Cross-domain (*mini*ImageNet → CUB). Cross-domain is a challenging scenario in few-shot classification due to the

			*mini*Imagenet → CUB
Method	Transductive	backbone	5-way 5-shot
MAML [Finn *et al.*, 2017]	✗	ResNet-18	51.34 ± 0.72
RelationNet [Sung *et al.*, 2018]	✗	ResNet-18	57.71 ± 0.73
MatchingNet [Vinyals *et al.*, 2016]	✗	ResNet-18	53.07 ± 0.74
ProtoNet [Snell *et al.*, 2017]	✗	ResNet-18	62.02 ± 0.70
Baseline [Chen *et al.*, 2019]	✗	ResNet-18	65.57 ± 0.70
Baseline++ [Chen *et al.*, 2019]	✗	ResNet-18	62.04 ± 0.76
SimpleShot [Wang *et al.*, 2019]	✗	ResNet-18	65.63
Negative-Cosine [Liu *et al.*, 2020a]	✗	ResNet-18	67.03 ± 0.76
S2M2-R [Mangla *et al.*, 2020]	✗	ResNet-18	70.44 ± 0.75
LaplacianShot [Ziko *et al.*, 2020]	✓	ResNet-18	66.33
TDO [Liu *et al.*, 2023]	✓	ResNet-18	68.82 ± 0.75
TIM [Boudiaf *et al.*, 2020]	✓	ResNet-18	71.00
Oblique Manifold [Qi *et al.*, 2021]	✓	ResNet-18	74.11
DIPO (ours)	✓	ResNet-18	**77.48 ± 0.19**

Table 3: The 5-way 5-shot accuracy (%) for cross-domain on *mini*ImageNet → CUB. The best results are highlighted in bold.

large domain shift between the base and novel domains. To show the effectiveness of our method in the cross-domain scenario, we conduct the tests on *mini*ImageNet → CUB. In particular, we follow the steps to train ResNet-18 described by [Chen *et al.*, 2019], where the base classes are drawn from *mini*ImageNet, and the novel classes are drawn from CUB. Table 3 shows the 5-way 5-shot results achieved using ResNet-18 as the feature extractor for a fair comparison. These results show that the proposed DIPO outperforms others by about 3% margins. Our method employs a mechanism to transfer the diversity of the top k relevant base classes to facilitate the recognition of novel classes. By selecting the number of relevant base classes k, it is adept at handling scenarios where the base and novel domains exhibit differences.

The class-unbalanced setting [Veilleux *et al.*, 2021] is presented in the supplementary material.

4.3 Ablation Study

We conduct the comparison experiments to show the effectiveness of the similarity-evaluating strategy and the density-driven strategy proposed in our DIPO. Since the similarity-evaluating strategy employs the Mahalanobis distance to measure the information gap between the labeled and unlabeled samples, we compare it with the cosine distance and Euclidean distance for better illustration. Results are presented in Table 4. The application of the density-driven strategy results in improvement gains of 14.16%, 12.92%, and 24.10% for the cosine distance, Euclidean distance, and similarity-evaluating strategy, respectively, in the 5-way 1-shot task. Since the density-driven strategy utilizes the query samples for transductive inference, it can derive a more accurate prototype than the initial prototype obtained solely from support samples. The first three rows and the last three rows in Table 4 compare the cosine distance, Euclidean distance, and similarity-evaluating strategy without and with the density-driven strategy, respectively. The similarity-evaluating strategy can adeptly capture the information gap between features compared to the other two distance metrics: 1) it amplifies the bias in the initial prototype, resulting in lower performance as shown in the first three rows; 2) it contributes to a greater improvement based on the higher-quality prototype as shown in the last three rows. The comparison results further demonstrate the effectiveness of our method.

				*mini*ImageNet	
cosine	Euclidean	SES	DDS	5-way 1-shot	5-way 5-shot
✓				63.12 ± 0.20	83.91 ± 0.13
	✓			62.81 ± 0.20	82.13 ± 0.14
		✓		60.31 ± 0.21	80.81 ± 0.15
✓			✓	77.28 ± 0.26	86.71 ± 0.14
	✓		✓	75.73 ± 0.27	84.64 ± 0.15
		✓	✓	**84.41 ± 0.24**	**90.70 ± 0.11**

Table 4: The ablation study for both the 5-way 1-shot and 5-way 5-shot settings on *mini*ImageNet with backbone WRN-28-10. cosine or Euclidean: cosine or Euclidean distance, the distance metric of labeled and unlabeled samples instead of the Mahalanobis distance in Equation 6 in the similarity-evaluating strategy. SES: similarity-evaluating strategy, without which we do not calculate the information gap using the the Mahalanobis distance. DDS: density-driven strategy, without which we just calculate the initial prototypes using support samples to be the final prototypes.

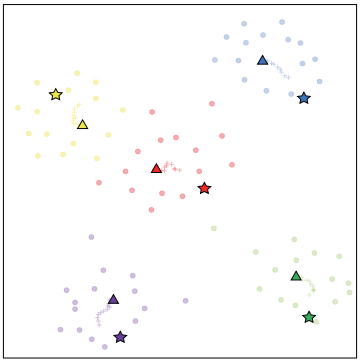

Figure 3: The t-SNE visualization of DIPO on a 5-way 1-shot task sampled from *mini*ImageNet. Different colors mean different classes. "★" means the initial prototypes, "o" means query samples, "+" means the refined prototypes at different iterations, and "Δ" means the final prototypes.

4.4 Visualization of DIPO

We visualize the prototypes refined by our DIPO for a 5-way 1-shot task on *mini*ImageNet by means of t-SNE [van der Maaten and Hinton, 2008], as shown in Figure 3. Note that the initial prototype is the support sample since there is only one support sample in each novel class. It can be seen that the initial prototypes are biased and far away from the class centers. By refining the prototypes in the direction of density growth, the prototypes progressively get closer to the class centers. This indicates that our DIPO effectively learns high-quality prototypes.

5 Conclusion

In this paper, we investigate the correlation between sample quality and density and propose a Density-driven Iterative Prototype Optimization (DIPO) to improve the quality of the prototypes. We show that the proposed method brings consistent performance improvements over multiple baselines and could reach or exceed the state-of-the-art performance on four benchmark datasets, including *mini*ImageNet, *tiered*ImageNet, CUB, and CIFAR-FS. The visualization of the refined prototypes by DIPO shows that our method can effectively learn high-quality prototypes.

Acknowledgments

This work was supported by the STI 2030-Major Projects under grant 2022ZD0208900, the Guangdong Basic and Applied Basic Research Foundation under grant 2024A151501052.

References

[Bertinetto *et al.*, 2018] Luca Bertinetto, Joao F Henriques, Philip HS Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018.

[Boudiaf *et al.*, 2020] Malik Boudiaf, Imtiaz Ziko, Jérôme Rony, José Dolz, Pablo Piantanida, and Ismail Ben Ayed. Information maximization for few-shot learning. *Advances in Neural Information Processing Systems*, 33:2445–2457, 2020.

[Chen *et al.*, 2019] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019.

[Chen *et al.*, 2020] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, Trevor Darrell, et al. A new meta-baseline for few-shot learning. *arXiv preprint arXiv:2003.04390*, 1(2):3, 2020.

[Chen *et al.*, 2021] Chaofan Chen, Xiaoshan Yang, Changsheng Xu, Xuhui Huang, and Zhe Ma. Eckpn: Explicit class knowledge propagation network for transductive few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6596–6605, 2021.

[Dang *et al.*, 2023] Zhuohang Dang, Minnan Luo, Chengyou Jia, Caixia Yan, Xiaojun Chang, and Qinghua Zheng. Counterfactual generation framework for few-shot learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[Dhillon *et al.*, 2019] Guneet S Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. *arXiv preprint arXiv:1909.02729*, 2019.

[Finn *et al.*, 2017] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

[Fukunaga and Hostetler, 1975] Keinosuke Fukunaga and Larry Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, 21(1):32–40, 1975.

[Gao *et al.*, 2018] Hang Gao, Zheng Shou, Alireza Zareian, Hanwang Zhang, and Shih-Fu Chang. Low-shot learning via covariance-preserving adversarial augmentation networks. *Advances in Neural Information Processing Systems*, 31, 2018.

[Hong *et al.*, 2020] 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)*, pages 1–6. IEEE, 2020.

[Hou *et al.*, 2019] Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification. *Advances in Neural Information Processing Systems*, 32, 2019.

[Hu *et al.*, 2020] Shell Xu Hu, Pablo G Moreno, Yang Xiao, Xi Shen, Guillaume Obozinski, Neil D Lawrence, and Andreas Damianou. Empirical bayes transductive meta-learning with synthetic gradients. *arXiv preprint arXiv:2004.12696*, 2020.

[Hu *et al.*, 2021] Yuqing Hu, Vincent Gripon, and Stéphane Pateux. Leveraging the feature distribution in transfer-based few-shot learning. In *International Conference on Artificial Neural Networks*, pages 487–499. Springer, 2021.

[Lazarou *et al.*, 2021] Michalis Lazarou, Tania Stathaki, and Yannis Avrithis. Iterative label cleaning for transductive and semi-supervised few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8751–8760, 2021.

[Li *et al.*, 2019] Wenbin Li, Jinglin Xu, Jing Huo, Lei Wang, Yang Gao, and Jiebo Luo. Distribution consistency based covariance metric networks for few-shot learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8642–8649, 2019.

[Li *et al.*, 2020] Kai Li, Yulun Zhang, Kunpeng Li, and Yun Fu. Adversarial feature hallucination networks for few-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13470–13479, 2020.

[Lichtenstein *et al.*, 2020] Moshe Lichtenstein, Prasanna Sattigeri, Rogerio Feris, Raja Giryes, and Leonid Karlinsky. Tafssl: Task-adaptive feature sub-space learning for few-shot classification. In *European Conference on Computer Vision*, pages 522–539. Springer, 2020.

[Liu *et al.*, 2020a] 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 *ECCV*, 2020.

[Liu *et al.*, 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*, pages 741–756. Springer, 2020.

[Liu *et al.*, 2022] Yuanwei Liu, Nian Liu, Xiwen Yao, and Junwei Han. Intermediate prototype mining transformer for few-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 35:38020–38031, 2022.

[Liu *et al.*, 2023] Xinyue Liu, Ligang Liu, Han Liu, and Xiaotong Zhang. Capturing the few-shot class distribution: Transductive distribution optimization. *Pattern Recognition*, 138:109371, 2023.

[Mangla *et al.*, 2020] 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 *WACV*, 2020.

[Qi *et al.*, 2021] Guodong Qi, Huimin Yu, Zhaohui Lu, and Shuzhao Li. Transductive few-shot classification on the oblique manifold. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8412–8422, 2021.

[Qiao *et al.*, 2019] Limeng Qiao, Yemin Shi, Jia Li, Yaowei Wang, Tiejun Huang, and Yonghong Tian. Transductive episodic-wise adaptive metric for few-shot learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3603–3612, 2019.

[Ravi and Larochelle, 2017] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International conference on learning representations*, 2017.

[Ren *et al.*, 2018] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018.

[Rizve *et al.*, 2021] 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*, pages 10836–10846, 2021.

[Russakovsky *et al.*, 2014] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[Rusu *et al.*, 2019] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *ICLR*, 2019.

[Snell *et al.*, 2017] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017.

[Sung *et al.*, 2018] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H.S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, 2018.

[Tao *et al.*, 2022] Ran Tao, Han Zhang, Yutong Zheng, and Marios Savvides. Powering finetuning in few-shot learning: Domain-agnostic bias reduction with selected sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8467–8475, 2022.

[Tian *et al.*, 2020] 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*, pages 266–282. Springer, 2020.

[Trosten *et al.*, 2023] Daniel J Trosten, Rwiddhi Chakraborty, Sigurd Løkse, Kristoffer Knutsen Wickstrøm, Robert Jenssen, and Michael C Kampffmeyer. Hubs and hyperspheres: Reducing hubness and improving transductive few-shot learning with hyperspherical embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7527–7536, 2023.

[Tukey, 1977] John W Tukey. *Exploratory data analysis*. Addison-Wesley Series in Behavioral Science. Addison-Wesley, Reading, MA, 1977.

[van der Maaten and Hinton, 2008] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 2008.

[Veilleux *et al.*, 2021] Olivier Veilleux, Malik Boudiaf, Pablo Piantanida, and Ismail Ben Ayed. Realistic evaluation of transductive few-shot learning. *Advances in Neural Information Processing Systems*, 34:9290–9302, 2021.

[Vinyals *et al.*, 2016] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, 2016.

[Wang *et al.*, 2019] 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.

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

[Yang *et al.*, 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*, pages 13390–13399, 2020.

[Yang *et al.*, 2021] Shuo Yang, Lu Liu, and Min Xu. Free lunch for few-shot learning: Distribution calibration. In *ICLR*, 2021.

[Zhang *et al.*, 2022] Baoquan Zhang, Xutao Li, Shanshan Feng, Yunming Ye, and Rui Ye. Metanode: Prototype optimization as a neural ode for few-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 9014–9021, 2022.

[Zhu and Koniusz, 2022] Hao Zhu and Piotr Koniusz. Ease: Unsupervised discriminant subspace learning for transductive few-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9078–9088, 2022.

[Zhu and Koniusz, 2023] Hao Zhu and Piotr Koniusz. Transductive few-shot learning with prototype-based label propagation by iterative graph refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23996–24006, 2023.

[Ziko *et al.*, 2020] Imtiaz Ziko, Jose Dolz, Eric Granger, and Ismail Ben Ayed. Laplacian regularized few-shot learning. In *International conference on machine learning*, pages 11660–11670. PMLR, 2020.