Few-Shot Fine-Grained Image Classification with Progressively Feature Refinement and Continuous Relationship Modeling

Zhen-Xiang Ma, Zhen-Duo Chen*, Tai Zheng, Xin Luo, Zixia Jia, Xin-Shun Xu

School of Software, Shandong University, Jinan, China
mazhenxiang0923@163.com, {chenzd.sdu, zt5369623, luoxin.lxin, jiazixia770}@gmail.com, xuxinshun@sdu.edu.cn

Abstract

Recently, a number of effective methods have been proposed to tackle the challenging task of Few-Shot Fine-Grained Image Classification (FS-FGIC). However, how to fully leverage the backbone network to discover and extract detailed features to generate more discriminative class prototypes, as well as how to accurately model the similarity relationship between query samples and the class prototypes, are still issues to be further considered. Therefore, we propose a novel progreSsively featUre refInement and conTinuous rElationship moDeling method, SUITED for short, to address these two issues existing in the State-of-the-Art FS-FGIC methods. Specifically, we design the Progressive Feature Refinement Module (PFRM) to fully exploit the backbone network's progressive feature extraction capabilities, forming multi-scale feature representations to further enhance discriminative features. Then, the Continuous Relationship Modeling Module (CRMM) is proposed to capture the dependencies between query samples and the corresponding class prototypes, achieving precise optimization of the distances among corresponding sample points in the feature space. We conducted extensive experiments on five fine-grained benchmark datasets, and the experimental results demonstrate that the proposed method is comprehensively ahead of the existing State-of-the-Art methods.

Introduction

With the continuous advancement of computer vision technology (He et al. 2016; Dosovitskiy et al. 2021), the task of Fine-Grained Image Classification (FGIC) has made unprecedented progress. FGIC aims to distinguish between visually similar subclasses under the same basic-level category, such as different breeds of dogs or different types of cars. However, rare category samples within fine-grained datasets are relatively scarce, and the cost of obtaining fine-grained annotations is expensive, which limits the general applicability of traditional FGIC methods. In contrast, humans can distinguish subtle differences between subclasses after learning with a very limited number of samples. Inspired by this, to enhance model generalization in data-scarce scenarios without relying on large-scale training data, researchers have shifted their focuses to Few-Shot Fine-Grained Image Classification (FS-FGIC), aiming to enable models to accurately recognize unseen novel samples with a few labeled samples by learning task-transfer knowledge from base classes, thus narrowing the gap with humans.

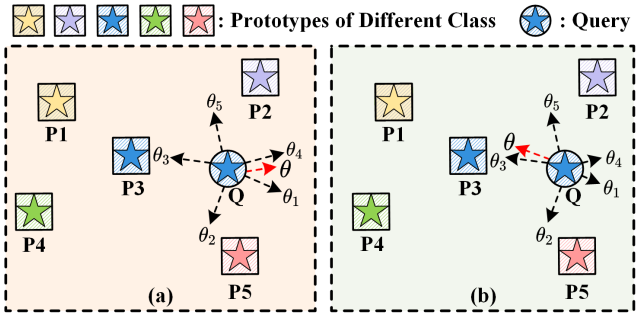

Figure 1: Comparison between the existing discrete modeling methods (a) and our proposed continuous modeling method (b) regarding Query optimization direction, where $\theta 1$ to $\theta 5$ are the optimization directions guided by **P1** to **P5**, and θ is the final optimization direction obtained by the vector sum of all these optimization directions.

The limited labeled data (support) for Few-Shot Learning (FSL) (Wertheimer, Tang, and Hariharan 2021) determines that the traditional supervised learning strategy cannot be effectively applied. During the past several years, the Prototype-based classification strategy proposed by ProtoNet (Snell, Swersky, and Zemel 2017) has gradually become the most representative FSL strategy. The simplicity and effectiveness of this strategy have inspired most subsequent methods, which have consistently maintained State-of-the-Art results for FSL and FS-FGIC tasks in recent years (Zhao et al. 2024; Ma et al. 2024; Lee, Moon, and Heo 2022). The basic idea of ProtoNet is to calculate the prototypes representing each class based on a limited number of support samples, and then determine the class of query samples by evaluating their similarity to these prototypes. Obviously, for the FS-FGIC task, two key points are crucial for the effectiveness of this strategy: Firstly, the prototypes should be sufficiently representative and discriminative of the categories, to effectively reflect the features that distin-

*Corresponding Author

guish a fine-grained category from other similar categories. Secondly, it is essential to accurately construct the similarity relationships between query samples and prototypes in the fine-grained dataset, and the model can be trained to effectively capture subtle similarities and differences.

The first key point is essentially a feature learning and representation issue, which is the primary focus of most existing methods. This involves designing better model architectures for fine-grained feature extraction, representation, and matching, enabling the model to extract sufficient discriminative information from input samples. However, how to fully leverage the detailed feature discovery and extraction capabilities of backbone to better serve subsequent FSL remains an open question, with no universally accepted ideas.

Compared to the problem of fine-grained feature learning, the second key point mentioned above has not received widespread attention. For model training, existing methods usually establish one-to-one relationships between query features and class prototypes according to labels. This simple similarity modeling strategy can only capture local relationships between sample pairs and lacks a global description of feature and semantic distributions, so it is difficult to ensure that the model trained with this objective can accurately capture the differences and similarities between fine-grained samples. Moreover, because there are only two types of relationship: similar and dissimilar, the query-prototype relationship defined in this way is discrete. So it is very difficult to accurately reflect the fine-grained feature distribution in the feature space. Figure 1a shows an example where we aim to ensure that after model training, the query Q is mapped close to its corresponding prototype **P3** while being distant from other prototypes. Clearly, the key to this task is guiding the model to effectively distinguish the subtle differences between highly similar fine-grained class prototypes (**P2**, **P3**, and **P5**), while the more distant prototypes, **P1** and **P4**, are relatively less important. However, the aforementioned one-to-one discrete relationship modeling can only guide model training by treating each sample pair in an isolated and equivalent way, leading to the optimization direction represented by the black arrows. Eventually, the overall optimization direction of the model may result in errors, as shown by the red arrow, thereby increasing the difficulty of model optimization.

To address the aforementioned issues, we propose a novel FS-FGIC method with progreSsively featUre refInement and conTinuous rElationship moDeling, SUITED for short. This method includes two modules: Progressively Feature Refinement Module (PFRM) and Continuous Relationship Modeling Module (CRMM). The PFRM is designed to enhance the discriminative feature representation for the fine-grained classes. Specifically, a top-down feature feedback mechanism is designed to progressively extract features from different layers with the guidance of higher-level features. This progressive mechanism, compared to standard multi-layer integration, more effectively maintains the complementarity and consistency among high, middle, and low-layer features. As a result, PFRM can comprehensively learn task-relevant discriminative features and mitigate overfitting issues resulted by FSL setting. Thereafter, the CRMM is proposed to accurately model the similarity dependence between query features and class prototype. The one-to-many continuous relationships among the prototype and the query samples are established based on high-layer features. On this basis, a continuous relationship mining network is designed and trained to capture the dependency between the query samples and the class prototype. Compared to the aforementioned one-to-one discrete strategy, the proposed CRMM can optimize the relationship between query **Q** and key prototypes (such as **P2**, **P3**, and **P5** in Figure 1b from a global perspective. During training, CRMM can better capture the subtle similarity relationships between fine-grained classes and samples, modify the optimization direction of the model (as shown by the red arrow in Figure 1b), and thus achieve better results after training.

Our main contributions could be summarized as follows:

- We analyze the key points and shortcomings of the mainstream frameworks adopted by the current FS-FGIC methods. We propose the SUITED, which provides targeted optimization solutions from the perspectives of feature learning and sample relationship modeling.
- The PFRM is designed to obtain multi-scale outputs from backbone, and a top-down feature feedback mechanism is designed to further enhance discriminative features.
- The CRMM is proposed to capture the dependency between the query and the corresponding class prototype, thereby achieving precise optimization of sample point distances in the feature space.
- Extensive experimental results on five few-shot fine-grained benchmark datasets indicate that our proposed method comprehensively outperforms the current State-of-the-Art methods.

Related Work

Few-Shot Learning

The goal of Few-Shot Learning is to train a model using the base classes that can easily adapt to the unseen novel classes. Existing FSL methods can be divided into meta-learning based methods (Finn, Abbeel, and Levine 2017; Lee et al. 2019), data-augmentation based methods (Chen et al. 2019; Tang et al. 2020; Hariharan and Girshick 2017), and metric-learning based methods (Sung et al. 2018; Vinyals et al. 2016), with the most popular being the metric learning-based methods. Meta-learning based methods aim to discover an effective gradient-based optimization strategy that enables quick adaptation to new tasks with only a few gradient updates. Data-augmentation based methods attempt to generate new samples or features from the trainable base classes to better address the issue of limited training samples. Metric-learning-based methods attempt to find an appropriate metric to measure the similarity between query features and support features. The most representative method is ProtoNet (Snell, Swersky, and Zemel 2017), which forms a prototype for each class using the mean of the support set for that class. Currently, most State-of-the-Art methods are based on ProtoNet.

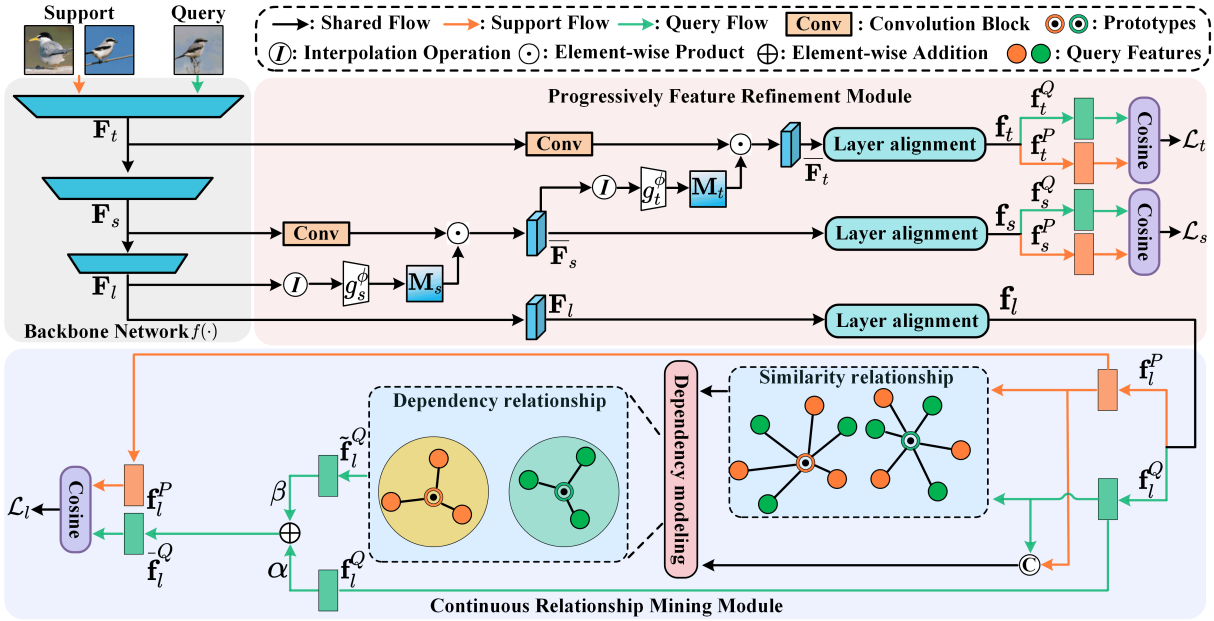

Figure 2: The overall architecture of the SUITED, consisting of the Backbone, the Progressively Feature Refinement Module (PFRM), and the Continuous Relationship Modeling Module (CRMM).

Few-Shot Fine-grained Image Classification

Traditional FSL methods cannot be directly applied to more challenging fine-grained datasets, as they fail to effectively extract fine-grained features. Therefore, they are unable to accurately identify fine-grained images with minor inter-class differences and large intra-class variations. Therefore, (Wei et al. 2019) defines the FS-FGIC. TDM (Lee, Moon, and Heo 2022) generates weights for each channel to better emphasize the discriminative fine-grained features. AIS-MLI (Zhao et al. 2024) introduces Angular ISotonic loss to guide the optimization direction of the feature space, enabling the network to converge more effectively.

Some existing methods attempt to establish relationship matching between query samples and support samples. HelixFormer (Zhang et al. 2022) learns cross-image semantic relationships through a double-helix multi-head attention mechanism. BiFRN (Wu et al. 2023) uses a bidirectional attention mechanism-based operation to reconstruct the relationship between query and support samples in high-level features. C2-Net (Ma et al. 2024) extracts fine-grained features from a cross-layer perspective and addresses feature mismatch issues from both channel and spatial dimensions. These methods establish one-to-one discrete relationships, which cannot accurately reflect the subtle similarities between fine-grained samples. In contrast, SUITED models the one-to-many relationship between prototypes and query samples in a continuous manner, achieving precise optimization of distances between sample points in the feature space.

Method

As shown in Figure 2, our proposed method consists of three components: The backbone network is used to extract deep convolution features of support and query samples. The PFRM obtains multi-scale feature representations while refining discriminative fine-grained features. The CRMM further explores the continuous relationships between query features and the corresponding class prototypes.

Problem Definition

Given the fine-grained dataset $D = \{(x_i, y_i), y_i \in Y)\}$, under the standard FSL settings, D will be divided into three parts: the training dataset $D_{base} = \{(x_i, y_i), y_i \in Y_{base})\}$, the validation dataset $D_{val} = \{(x_i, y_i), y_i \in Y_{val})\}$, and the test dataset $D_{novel} = \{(x_i, y_i), y_i \in Y_{novel})\}$, where $Y_{base} \cap Y_{val} \cap Y_{novel} = \emptyset$. The training and testing phases are composed of episodes. Specifically, each episode consists of N randomly sampled classes, with each class having K labeled support samples and U unlabeled query samples, forming an "N-way K-shot" classification task.

Progressively Feature Refinement Module

Relying solely on high-layer features of the single-scale cannot capture enough fine-grained information, making it difficult to discriminate between classes with minor inter-class differences in fine-grained images. Moreover, under the FSL settings, the limited number of training samples may lead the model to focus on task-irrelevant information, thus affecting the performance of the FS-FGIC task. To solve this problem, we design the Progressively Feature Refinement Module (PFRM) to obtain multi-scale feature representations from the backbone network.

Given an input sample $\mathcal{I}$, we first obtain the output of the last layer $\mathbf{F}_l \in \mathbb{R}^{C^l \times H^l \times W^l}$, the second-to-last layer $\mathbf{F}_s \in$

$\mathbb{R}^{C^s \times H^s \times W^s}$, and the third-to-last layer $\mathbf{F}_t \in \mathbb{R}^{C^t \times H^t \times W^t}$ from the backbone network f:

$$\mathbf{F}_l, \mathbf{F}_s, \mathbf{F}_t = f(\mathcal{I}). \tag{1}$$

Since the feature representation ability of the first layer output is too weak, it is not used in this paper.

Next, considering that the features extracted from adjacent layers are complementary and that high-layer features are gradually constructed from low-layer features, we design a top-down feature feedback mechanism. Specifically, we use interpolation operation I to adjust the feature map size of $\mathbf{F}_l$ to match the feature map size of $\mathbf{F}_s$, which can be formulated as

$$\mathbf{F}'_l = I(\mathbf{F}_l) \in \mathbb{R}^{C^l \times H^s \times W^s}. \tag{2}$$

Thereafter, we use a sub-network g_s^ϕ that includes two convolutional layers to generate high-level features for guiding the feedback enhancement map $\mathbf{M}_s$ of the mid-level features as follows,

$$\mathbf{M}_s = g_s^\phi(\mathbf{F}'_l) \in \mathbb{R}^{1 \times H^s \times W^s}. \tag{3}$$

Next, $\mathbf{M}_s$ is used to adjust the mid-level features element-wise, enhancing local discriminative features while weakening task-irrelevant information as follows,

$$\overline{\mathbf{F}}_s = \mathbf{M}_s \odot Conv_s(\mathbf{F}_s) \in \mathbb{R}^{C^s \times H^s \times W^s}, \tag{4}$$

where $Conv_s$ is a convolution block.

After obtaining the refined mid-layer features $\overline{\mathbf{F}}_s$, we progressively use a similar process to generate the feedback enhancement map $\mathbf{M}_t$ for the low-layer features $\mathbf{F}_t$ using $\overline{\mathbf{F}}_s$, and then obtain the refined low-level features $\overline{\mathbf{F}}_t$, which can be formulated as

$$\mathbf{F}'_s = I(\overline{\mathbf{F}}_s) \in \mathbb{R}^{C^s \times H^t \times W^t}, \tag{5}$$

$$\mathbf{M}_t = g_t^\phi(\mathbf{F}'_s) \in \mathbb{R}^{1 \times H^t \times W^t}, \tag{6}$$

$$\overline{\mathbf{F}}_t = \mathbf{M}_t \odot Conv_s(\mathbf{F}_t) \in \mathbb{R}^{C^t \times H^t \times W^t}. \tag{7}$$

To reduce the differences between features from different layers, we further design layer alignment sub-networks to ensure spatial and semantic consistency across different layers, which allows the model to generate more category-representative prototypes under the extremely limited FSL training sample setting. Features $\mathbf{f}_l \in \mathbb{R}^{C^l}$, $\mathbf{f}_s \in \mathbb{R}^{C^s}$, and $\mathbf{f}_t \in \mathbb{R}^{C^t}$ can be obtained as follows,

$$\mathbf{f}_l = MLP_l(\sigma(g_l^\theta(\mathbf{F}_l))), \tag{8}$$

$$\mathbf{f}_s = MLP_s(\sigma(g_s^\theta(\overline{\mathbf{F}}_s))), \tag{9}$$

$$\mathbf{f}_t = MLP_t(\sigma(g_t^\theta(\overline{\mathbf{F}}_t))), \tag{10}$$

where g_l^θ, g_s^θ, and g_t^θ are composed of a convolution block respectively, σ is global max pooling, and MLP represents the fully connected block.

Finally, we separate the query features $(\mathbf{f}_l^Q, \mathbf{f}_s^Q, \mathbf{f}_t^Q)$ and support features $(\mathbf{f}_l^S, \mathbf{f}_s^S, \mathbf{f}_t^S)$ from the features $\mathbf{f}_l$, $\mathbf{f}_s$, and $\mathbf{f}_t$, respectively. Then, following ProtoNet, we generate the corresponding prototype $(\mathbf{f}_l^P, \mathbf{f}_s^P, \mathbf{f}_t^P)$ for each class as follows,

$$\mathbf{f}_{i,a}^P = \frac{1}{K} \sum_{j=1}^{K} \mathbf{f}_{i,j,a}^S, \tag{11}$$

where $a = \{l, s, t\}$, i represents the i-th class, and j represents the j-th image.

In this module, we progressively explore the feature discovery and extraction capabilities of the backbone under the setting with very limited training samples, forming multi-scale feature representations. This not only strengthens the network's ability to extract task-relevant discriminative features but also reduces the network's focus on task-irrelevant information. Consequently, more category-representative and discriminative prototypes can be obtained through PFRM, thereby alleviating the overfitting problem.

Continuous Relationship Modeling Module

After obtaining prototypes with sufficient category representation, how to correctly model the similarity relationship between query features and the corresponding prototype is the next issue. Traditional one-to-one modeling methods first generate a corresponding support feature for each query feature to form sample pairs, and then perform relationship matching across samples. In contrast, we propose the Continuous Relationship Modeling Module (CRMM), which establishes a continuous one-to-many relationship between the prototype and query features, thereby achieving precise optimization of the distances between corresponding sample points in the feature space.

In this module, we only use $\mathbf{f}_l^Q$ and $\mathbf{f}_l^P$ because high-layer features contain rich semantic information, and the corresponding prototypes have stronger category representation abilities. We first divide the prototypes and query features into two non-overlapping groups according to the corresponding quantities as follows,

$$\begin{aligned} &\text{Group}_P = \{0, 1, \ldots, N-1\}, \\ &\text{Group}_Q = \{N, N+1, \ldots, N + N \times U - 1\}. \end{aligned} \tag{12}$$

Then we establish a one-to-many relationship index matrix $\mathbf{E}$ between the prototypes Group_P and the query features Group_Q, which can be formulated as

$$\mathbf{E} = \left\{ (m, n) \mid m \in \text{Group}_P, n \in \text{Group}_Q \right\}. \tag{13}$$

Next, we establish the continuous one-to-many relationship between each prototype and all query features. Specifically, we establish the similarity relation matrix $\mathbf{R}$ based on the relationship index matrix $\mathbf{E}$, where the similarity relationship between the m-th prototype and the n-th query feature is calculated as follows,

$$\mathbf{R}_{m,n} = \begin{cases} \frac{\mathbf{f}_{m,l}^P \cdot \mathbf{f}_{n,l}^Q}{\|\mathbf{f}_{m,l}^P\| \|\mathbf{f}_{n,l}^Q\|} & \text{if } (m,n) \in E \\ 0 & \text{otherwise} \end{cases}. \tag{14}$$

Inspired by Graph Convolutional Network (GCN) (Kipf and Welling 2017), we consider the prototypes and query features as nodes and the constructed similarity relationships

Dataset	N_{train}	N_{val}	N_{test}
CUB-200-2011	100	50	50
Stanford-Dogs	70	20	30
Stanford-Cars	130	17	49
meta-iNat	908	-	227
tiered meta-iNat	781	-	354

Table 1: The splitting way of datasets, N_{train}, N_{val}, and N_{test} represent the number of classes for training, validation, and test, respectively.

as edges between the nodes. We then design a GCN-based dependency modeling sub-network to capture the continuous dependencies between the query features and the corresponding class prototypes, which can be formulated as

$$\tilde{\mathbf{C}} = ReLU\left(\widetilde{\mathbf{D}}^{-\frac{1}{2}}(\mathbf{R}+\mathbf{I})\widetilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{C}W\right) \in \mathbb{R}^{(N+N\times U)\times C^l}, \tag{15}$$

$$\mathbf{C} = \left[\mathbf{f}_l^P, \mathbf{f}_l^Q\right] \in \mathbb{R}^{(N+N\times U)\times C^l}, \tag{16}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{D} = \sum_n \mathbf{R}_{m,n}$ is the normalized diagonal matrix, W is the learnable parameters, and $[\cdot,\cdot]$ is the concatenation operation.

Finally, we separate the continuously modeled $\tilde{\mathbf{f}}_l^Q$ from $\tilde{\mathbf{C}}$ to adjust the original query features $\mathbf{f}_l^Q$ in the feature space, achieving precise optimization of the distances to the corresponding class prototype, which can be formulated as

$$\bar{\mathbf{f}}_l^Q = \alpha\mathbf{f}_l^Q + \beta\tilde{\mathbf{f}}_l^Q, \tag{17}$$

where α and β are learnable weight parameters.

The above process allows the model to effectively capture subtle similarities and differences in fine-grained data, maintaining strong generalization performance on novel unseen classes. It should be noted that the CRMM guides the model to achieve precise optimization of sample point distances during the training process, and removing the CRMM during testing process has little impact on performance.

After the PFRM and CRMM, we can obtain the final (prototype, query) outputs: $(\mathbf{f}_l^P, \bar{\mathbf{f}}_l^Q)$, $(\mathbf{f}_s^P, \mathbf{f}_s^Q)$, and $(\mathbf{f}_t^P, \mathbf{f}_t^Q)$.

Overall Objectives

For a clearer representation, we simplify the final outputs of three layers to $(\mathbf{f}_a^P, \mathbf{f}_a^Q)$, where $a = \{l, s, t\}$, with $\bar{\mathbf{f}}_l^Q$ simplified to $\mathbf{f}_l^Q$. In the episodic training strategy, the loss function for an N-way K-shot task is calculated as follows,

$$\mathcal{L}_a = -\frac{1}{N}\frac{1}{U}\sum_{i=1}^{N}\sum_{j=1}^{U} log\frac{\exp\left(\tau_a\mathcal{S}\left(\mathbf{f}_{i,j,a}^Q, \mathbf{f}_{i,a}^P\right)\right)}{\sum_{p=1}^{N}\exp\left(\tau_a\mathcal{S}\left(\mathbf{f}_{i,j,a}^Q, \mathbf{f}_{p,a}^P\right)\right)}, \tag{18}$$

where $\mathcal{S}(\cdot)$ represents the Cosine similarity and τ_a are learnable temperature parameters.

In the end, the total loss $\mathcal{L}_{total}$ is calculated as follows,

$$\mathcal{L}_{total} = \mathcal{L}_l + \mathcal{L}_s + \mathcal{L}_t. \tag{19}$$

Model	Backbone	CUB	
		1-shot	5-shot
ProtoNet (NIPS-17)	Conv-4	59.95	76.01
BSNet (TIP-21)	Conv-4	55.81	76.34
PoseNorm (CVPR-20)	Conv-4	64.17	81.96
FEAT (CVPR-20)	Conv-4	68.87	82.90
FRN (CVPR-21)	Conv-4	69.45	85.16
OLSA (MM-21)	Conv-4	73.07	86.24
DAN (AAAI-22)	Conv-4	72.89	86.60
FRN + TDM (CVPR-22)	Conv-4	71.37	86.45
AGPF (PR-22)	Conv-4	74.03	86.54
PaCL (MM-22)	Conv-4	74.04	88.75
BiFRN† (AAAI-23)	Conv-4	74.98	89.14
C2-Net (AAAI-24)	Conv-4	78.66	89.43
Ours	Conv-4	**79.73**	**90.05**
P-Transfer (AAAI-21)	ResNet-12	73.16	88.32
DeepEMD (CVPR-20)	ResNet-12	75.65	88.69
BML (CVPR-21)	ResNet-12	76.16	90.32
OLSA (MM-21)	ResNet-12	77.77	89.87
FRN (CVPR-21)	ResNet-12	83.55	92.92
ISC (TIP-22)	ResNet-12	73.72	87.51
AGPF (PR-22)	ResNet-12	78.73	89.77
PaCL (MM-22)	ResNet-12	77.80	92.07
FRN + TDM (CVPR-22)	ResNet-12	84.36	93.37
BFSA (TSCVT-23)	ResNet-12	82.27	90.76
FGFL (ICCV-23)	ResNet-12	80.77	92.01
RENet-ventral (AAAI-23)	ResNet-12	83.33	92.97
BiFRN† (AAAI-23)	ResNet-12	82.07	92.11
MLI (TIP-24)	ResNet-12	85.94	93.50
C2-Net (AAAI-24)	ResNet-12	83.31	92.18
Ours	ResNet-12	**86.02**	**94.13**

Table 2: Performance (%) on the CUB (using raw images) dataset. "†" indicates our implementation based on the public code under the same splitting way of datasets. The highest results are highlighted.

Experiments

Dataset

We evaluated the proposed method on five fine-grained few-shot benchmark datasets: CUB-200-2011 (Wah et al. 2011), Stanford Dogs (Khosla et al. 2011), Stanford Cars (Krause et al. 2013), meta-iNat (Horn et al. 2018; Wertheimer and Hariharan 2019), and tiered meta-iNat (Wertheimer and Hariharan 2019). We follow the dataset splitting way most commonly used by current State-of-the-Art methods (Ma et al. 2024; Zhu, Liu, and Jiang 2020), as detailed in Table 1.

Implementation Details

In this paper, we adopt two backbone networks consistent with all current State-of-the-Art methods for a fair comparison, namely Conv-4 and ResNet-12. **Episodic Meta-training Details:** For both Conv-4 and ResNet-12, the size of the input images are resized to 84×84. We apply data

Model	Backbone	Stanford-Dogs		Stanford-Cars	
		1-shot	5-shot	1-shot	5-shot
ProtoNet (NIPS-17)	Conv-4	40.81 ± 0.83	61.58 ± 0.71	36.51 ± 0.74	62.14 ± 0.76
BSNet (TIP-21)	Conv-4	43.13 ± 0.85	62.61 ± 0.73	44.56 ± 0.83	63.72 ± 0.78
FRN (CVPR-21)	Conv-4	49.37 ± 0.20	67.13 ± 0.17	58.90 ± 0.22	79.65 ± 0.15
OLSA (MM-21)	Conv-4	55.53 ± 0.45	71.68 ± 0.36	70.13 ± 0.48	84.29 ± 0.31
DAN (AAAI-22)	Conv-4	59.81 ± 0.50	77.19 ± 0.35	70.21 ± 0.50	85.55 ± 0.31
AGPF (PR-22)	Conv-4	60.89 ± 0.98	78.14 ± 0.62	78.14 ± 0.84	87.42 ± 0.57
PaCL (MM-22)	Conv-4	59.76 ± 0.70	77.50 ± 0.48	72.21 ± 0.68	88.02 ± 0.36
HelixFormer (MM-22)	Conv-4	59.81 ± 0.50	73.40 ± 0.36	75.46 ± 0.37	89.68 ± 0.25
LCCRN (TCSVT-23)	Conv-4	-	-	71.62 ± 0.21	86.41 ± 0.12
BiFRN† (AAAI-23)	Conv-4	61.39 ± 0.23	78.86 ± 0.15	76.22 ± 0.20	90.66 ± 0.11
MLI (TIP-24)	Conv-4	63.13 ± 0.51	78.34 ± 0.35	-	-
C2-Net (AAAI-24)	Conv-4	66.42 ± 0.50	81.23 ± 0.34	81.29 ± 0.45	91.08 ± 0.26
Ours	Conv-4	**68.67 ± 0.51**	**82.24 ± 0.32**	**82.21 ± 0.44**	**92.39 ± 0.24**
BSNet (TIP-21)	ResNet-12	61.95 ± 0.97	79.62 ± 0.63	71.07 ± 1.03	88.38 ± 0.62
OLSA (MM-21)	ResNet-12	64.15 ± 0.49	78.28 ± 0.32	77.03 ± 0.46	88.85 ± 0.46
AGPF (PR-22)	ResNet-12	72.34 ± 0.86	84.02 ± 0.57	85.34 ± 0.74	94.79 ± 0.35
HelixFormer (MM-22)	ResNet-12	65.92 ± 0.49	80.65 ± 0.36	79.40 ± 0.43	92.26 ± 0.15
BFSA (TSCVT-23)	ResNet-12	69.58 ± 0.50	82.59 ± 0.33	88.93 ± 0.38	95.20 ± 0.20
LCCRN (TCSVT-23)	ResNet-12	-	-	87.04 ± 0.17	96.09 ± 0.07
BiFRN† (AAAI-23)	ResNet-12	72.54 ± 0.22	85.86 ± 0.13	88.43 ± 0.17	96.34 ± 0.07
MLI (TIP-24)	ResNet-12	76.32 ± 0.47	88.25 ± 0.27	-	-
C2-Net (AAAI-24)	ResNet-12	75.50 ± 0.49	87.65 ± 0.28	88.96 ± 0.37	95.16 ± 0.20
Ours	ResNet-12	**76.55 ± 0.47**	**88.86 ± 0.27**	**89.97 ± 0.36**	**96.53 ± 0.16**

Table 3: Performance (%) on the Stanford Dogs and Stanford Cars datasets.

Model	meta-iNat		tiered meta-iNat	
	1-shot	5-shot	1-shot	5-shot
ProtoNet (NIPS-17)	53.78	73.80	35.47	54.85
CTX (NIPS-20)	60.03	78.80	36.83	60.84
DeepEMD (CVPR-20)	54.48	68.36	36.05	48.55
FRN (CVPR-21)	61.98	80.04	43.95	63.45
FRN + TDM (CVPR-22)	63.97	81.60	44.05	62.91
MCL (CVPR-22)	64.66	81.31	44.08	64.61
BiFRN† (AAAI-23)	66.07	83.30	46.64	66.46
MLI (TIP-24)	68.42	82.46	48.97	67.15
C2-Net (AAAI-24)	71.47	85.47	49.04	67.25
Ours	**74.72**	**87.44**	**51.70**	**70.43**

Table 4: Performance (%) on the meta-iNat and tiered meta-iNat datasets.

augmentation techniques consistent with existing methods for all benchmark datasets, including random crop, horizontal flip, and color jitter. During model training, we used the SGD optimizer with Nesterov momentum of 0.9. The initial learning rate is set to 0.1, and the weight decay is set to 5e-4.

Episodic Meta-testing Details: We apply the standard 5-way 1-shot and 5-way 5-shot settings, with 15 query images per class. The model that performs best on the validation set is saved for testing. For all experiments, we report results with a 95% confidence interval for 2,000 test episodes.

Comparison to State-of-the-Art Methods

In this section, we conduct comparative experiments between our proposed method and the current State-of-the-Art methods. On the typical fine-grained datasets, including CUB (using raw images), Stanford Dogs, and Stanford Cars, the experimental results are shown in Tables 2 and 3. On more challenging datasets, meta-iNat and tiered meta-iNat, the experimental results are shown in Table 4.

The experimental results reveal the following points: Firstly, the proposed method achieves comprehensive performance superiority compared to other State-of-the-Art methods across all datasets, demonstrating the effectiveness of the proposed method. Secondly, compared to methods that model sample relationships on a one-to-one basis (HelixFormer, BiFRN, and C2-Net), SUITED shows a significant performance advantage in all settings. This indicates that by establishing a one-to-many continuous relationship, our method allows for more precise optimization of the distances between sample points in the feature space, thereby enhancing the performance of the FS-FGIC task. Finally, our proposed method demonstrates even more significant performance advantages on the more challenging meta-iNat and tiered meta-iNat datasets, further proving that SUITED has strong generalization capabilities on unseen novel classes

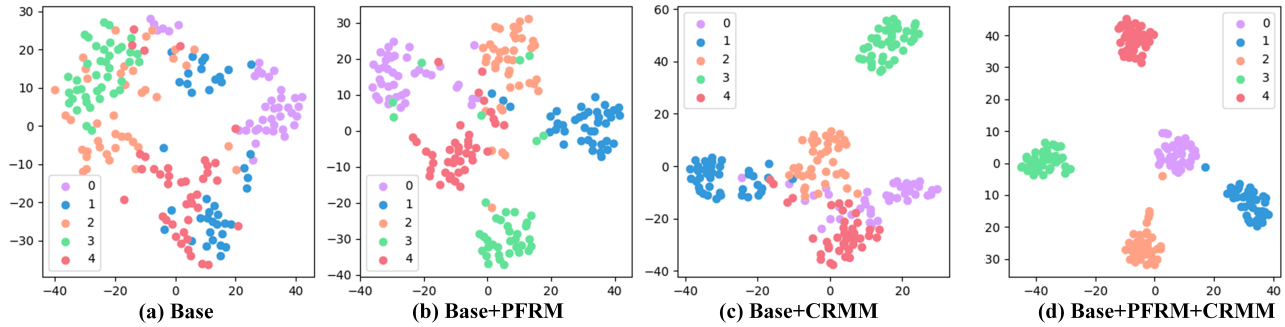

Figure 3: t-SNE visualization results of the embedding space using Conv-4 backbone on the CUB dataset.

Backbone	PFRM	CRMM	CUB		Stanford-Cars	
			1-shot	5-shot	1-shot	5-shot
✓			68.87	82.68	67.08	82.45
✓	✓		77.54	88.27	78.34	89.56
✓		✓	73.98	86.95	73.27	87.40
✓	✓	✓	**79.73**	**90.05**	**82.21**	**92.39**

Table 5: Ablation study of submodules using Conv-4 backbone on the CUB and Stanford-Cars datasets.

Model	CUB		Stanford-Cars	
	1-shot	5-shot	1-shot	5-shot
Baseline	77.54	88.27	78.34	89.56
CRMM (P to P + Q)	77.85	88.47	80.39	91.09
CRMM (P + Q to P + Q)	78.61	89.18	81.29	91.54
CRMM (P to Q)	**79.73**	**90.05**	**82.21**	**92.39**

Table 6: The experimental results of variants of CMRR using Conv-4 backbone. **Baseline** is composed of **Backbone** and **PFRM**, "**P**" denotes prototypes, "**Q**" denotes query features, and **CMRR (A to B)** represents the establishment of a one-to-many continuous relationship from **A** to **B**.

with very limited training samples.

Ablation Study and Analysis

Ablation Study of Submodules Table 5 reports the performance impact of the proposed PFRM and CRMM. The experimental results show that PFRM has significantly improved performance, and CRMM also contributes to performance improvement, though not as markedly as PFRM. The best performance is achieved when both modules are used together. This indicates that having sufficient discriminative features is a prerequisite for addressing the FS-FGIC task, but it is not enough to achieve State-of-the-Art performance, correct optimization of distances between corresponding sample points in the feature space is also necessary, which further demonstrates the effectiveness of the proposed two modules.

Variants of the CMRR We conducted additional experiments, as shown in Table 6, to further demonstrate that the proposed CMRR module is the ideal module for solving the FS-FGIC task. The experimental results indicate that capturing a one-to-many continuous similarity relationship from prototypes to query features achieves optimal performance.

Visualization of Feature Space

To visually demonstrate the effectiveness of the proposed method, we show the t-SNE visualization results in Figure 3. We randomly select five novel classes from the test set, and each class contains 40 random query samples.

It can be observed that the baseline classification performance is poor, as shown in Figure 3(a). When only PFRM is added, the introduction of multi-scale discriminative features makes the boundaries of each class relatively clear, but the samples of the same class are not clustered together, which affects the correct classification of samples near the boundaries, as shown in Figure 3(b). When only CMRR is added, samples of the same class are clustered, but the lack of sufficient fine-grained feature guidance results in blurred boundaries between classes, affecting the performance, as shown in Figure 3(c). Combining PFRM and CMRR not only clusters samples of the same class but also increases the distances between classes, making the boundaries particularly clear, as shown in Figure 3(d). This further demonstrates the effectiveness of the proposed method.

Conclusion

In this paper, we propose a novel progreSsively featUre refInement and conTinuous rElationship moDeling method, SUITED for short, to solve the challenging FS-FGIC task. The SUITED consists of two main modules: The PFRM obtains multi-scale feature representations while further refining discriminative fine-grained features. The CRMM effectively models the continuous relationship between query features and the corresponding class prototypes, achieving precise optimization of sample point distances in the feature space. Extensive experimental results on five benchmark datasets demonstrate that SUITED consistently outperforms existing State-of-the-Art methods.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62202272, Grant 62172256, and Grant 62202278, in part by Natural Science Foundation of Shandong Province under Grant ZR2019ZD06, and in part by the Young Scholars Program of Shandong University.

References

Chen, Z.; Fu, Y.; Zhang, Y.; Jiang, Y.; Xue, X.; and Sigal, L. 2019. Multi-Level Semantic Feature Augmentation for One-Shot Learning. *TIP*, 28(9): 4594–4605.

Cheng, H.; Yang, S.; Zhou, J. T.; Guo, L.; and Wen, B. 2023. Frequency Guidance Matters in Few-Shot Learning. In *ICCV*, 11814–11824.

Cheng, J.; Hao, F.; Liu, L.; and Tao, D. 2022. Imposing Semantic Consistency of Local Descriptors for Few-Shot Learning. *TIP*, 31: 1587–1600.

Doersch, C.; Gupta, A.; and Zisserman, A. 2020. CrossTransformers: spatially-aware few-shot transfer. In *NeurIPS*, 21981–21993.

Dong, L.; Zhai, W.; and Zha, Z. 2023. Exploring Tuning Characteristics of Ventral Stream's Neurons for Few-Shot Image Classification. In *AAAI*, 534–542.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *ICML*, 1126–1135.

Hariharan, B.; and Girshick, R. B. 2017. Low-Shot Visual Recognition by Shrinking and Hallucinating Features. In *ICCV*, 3037–3046.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *CVPR*, 770–778.

Horn, G. V.; Aodha, O. M.; Song, Y.; Cui, Y.; Sun, C.; Shepard, A.; Adam, H.; Perona, P.; and Belongie, S. J. 2018. The INaturalist Species Classification and Detection Dataset. In *CVPR*, 8769–8778.

Khosla, A.; Jayadevaprakash, N.; Yao, B.; and Fei-Fei, L. 2011. Novel Dataset for Fine-Grained Image Categorization. In *CVPR Workshop*, 806–813.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3D Object Representations for Fine-Grained Categorization. In *ICCV Workshops*, 554–561.

Lee, K.; Maji, S.; Ravichandran, A.; and Soatto, S. 2019. Meta-Learning With Differentiable Convex Optimization. In *CVPR*, 10657–10665.

Lee, S. B.; Moon, W.; and Heo, J. 2022. Task Discrepancy Maximization for Fine-grained Few-Shot Classification. In *CVPR*, 5321–5330.

Li, X.; Song, Q.; Wu, J.; Zhu, R.; Ma, Z.; and Xue, J. 2023. Locally-Enriched Cross-Reconstruction for Few-Shot Fine-Grained Image Classification. *TCSVT*, 33(12): 7530–7540.

Li, X.; Wu, J.; Sun, Z.; Ma, Z.; Cao, J.; and Xue, J. 2021. BSNet: Bi-Similarity Network for Few-shot Fine-grained Image Classification. *TIP*, 30: 1318–1331.

Liu, Y.; Zhang, W.; Xiang, C.; Zheng, T.; Cai, D.; and He, X. 2022. Learning to Affiliate: Mutual Centralized Learning for Few-shot Classification. In *CVPR*, 14391–14400.

Ma, Z.; Chen, Z.; Zhao, L.; Zhang, Z.; Luo, X.; and Xu, X. 2024. Cross-Layer and Cross-Sample Feature Optimization Network for Few-Shot Fine-Grained Image Classification. In *AAAI*, 4136–4144.

Shen, Z.; Liu, Z.; Qin, J.; Savvides, M.; and Cheng, K. 2021. Partial Is Better Than All: Revisiting Fine-tuning Strategy for Few-shot Learning. In *AAAI*, 9594–9602.

Snell, J.; Swersky, K.; and Zemel, R. S. 2017. Prototypical Networks for Few-shot Learning. In *NeurIPS*, 4077–4087.

Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P. H. S.; and Hospedales, T. M. 2018. Learning to Compare: Relation Network for Few-Shot Learning. In *CVPR*, 1199–1208.

Tang, H.; Li, Z.; Peng, Z.; and Tang, J. 2020. BlockMix: Meta Regularization and Self-Calibrated Inference for Metric-Based Meta-Learning. In *ACM MM*, 610–618.

Tang, H.; Yuan, C.; Li, Z.; and Tang, J. 2022. Learning attention-guided pyramidal features for few-shot fine-grained recognition. *Pattern Recognition*, 130: 108792.

Tang, L.; Wertheimer, D.; and Hariharan, B. 2020. Revisiting Pose-Normalization for Fine-Grained Few-Shot Recognition. In *CVPR*, 14340–14349.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; and Wierstra, D. 2016. Matching Networks for One Shot Learning. In *NeurIPS*, 3630–3638.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The Caltech-UCSD Birds-200-2011 Dataset.

Wang, C.; Fu, H.; and Ma, H. 2022. PaCL: Part-level Contrastive Learning for Fine-grained Few-shot Image Classification. In *ACM MM*, 6416–6424.

Wei, X.; Wang, P.; Liu, L.; Shen, C.; and Wu, J. 2019. Piecewise Classifier Mappings: Learning Fine-Grained Learners for Novel Categories With Few Examples. *TIP*, 28(12): 6116–6125.

Wertheimer, D.; and Hariharan, B. 2019. Few-Shot Learning With Localization in Realistic Settings. In *CVPR*, 6558–6567.

Wertheimer, D.; Tang, L.; and Hariharan, B. 2021. Few-Shot Classification With Feature Map Reconstruction Networks. In *CVPR*, 8012–8021.

Wu, J.; Chang, D.; Sain, A.; Li, X.; Ma, Z.; Cao, J.; Guo, J.; and Song, Y. 2023. Bi-directional Feature Reconstruction Network for Fine-Grained Few-Shot Image Classification. In *AAAI*, 2821–2829.

Wu, Y.; Zhang, B.; Yu, G.; Zhang, W.; Wang, B.; Chen, T.; and Fan, J. 2021. Object-aware Long-short-range Spatial Alignment for Few-Shot Fine-Grained Image Classification. In *ACM MM*, 107–115.

Xu, S.; Zhang, F.; Wei, X.; and Wang, J. 2022. Dual Attention Networks for Few-Shot Fine-Grained Recognition. In *AAAI*, 2911–2919.

Ye, H.; Hu, H.; Zhan, D.; and Sha, F. 2020. Few-Shot Learning via Embedding Adaptation With Set-to-Set Functions. In *CVPR*, 8805–8814.

Zha, Z.; Tang, H.; Sun, Y.; and Tang, J. 2023. Boosting Few-Shot Fine-Grained Recognition With Background Suppression and Foreground Alignment. *TCSVT*, 33(8): 3947–3961.

Zhang, B.; Yuan, J.; Li, B.; Chen, T.; Fan, J.; and Shi, B. 2022. Learning Cross-Image Object Semantic Relation in Transformer for Few-Shot Fine-Grained Image Classification. In *ACM MM*, 2135–2144.

Zhang, C.; Cai, Y.; Lin, G.; and Shen, C. 2020. DeepEMD: Few-Shot Image Classification With Differentiable Earth Mover's Distance and Structured Classifiers. In *CVPR*, 12200–12210.

Zhao, L.; Chen, Z.; Ma, Z.; Luo, X.; and Xu, X. 2024. Angular Isotonic Loss Guided Multi-Layer Integration for Few-Shot Fine-Grained Image Classification. *TIP*, 33: 3778–3792.

Zhou, Z.; Qiu, X.; Xie, J.; Wu, J.; and Zhang, C. 2021. Binocular Mutual Learning for Improving Few-shot Classification. In *ICCV*, 8382–8391.

Zhu, Y.; Liu, C.; and Jiang, S. 2020. Multi-attention Meta Learning for Few-shot Fine-grained Image Recognition. In *IJCAI*, 1090–1096.