Evolving and Regularizing Meta-Environment Learner for Fine-Grained Few-Shot Class-Incremental Learning

Li-Jun Zhao[1], Zhen-Duo Chen[1,*], Yongxin Wang[2], Xin Luo[1], Xin-Shun Xu[1]
[1]School of Software, Shandong University, China
[2]School of Computer and Artificial Intelligence, Shandong Jianzhu University, China
lj_zhao1028@163.com, chenzd.sdu@gmail.com, yxinwang@hotmail.com
luoxin.lxin@gmail.com, xuxinshun@sdu.edu.cn

Abstract

Recently proposed Fine-Grained Few-Shot Class-Incremental Learning (FG-FSCIL) offers a practical and efficient solution for enabling models to incrementally learn new fine-grained categories under limited data conditions. However, existing methods still settle for the fine-grained feature extraction capabilities learned from the base classes. Unlike conventional datasets, fine-grained categories exhibit subtle inter-class variations, naturally fostering latent synergy among sub-categories. Meanwhile, the incremental learning framework offers an opportunity to progressively strengthen this synergy by incorporating new sub-category data over time. Motivated by this, we theoretically formulate the FSCIL problem and derive a generalization error bound within a shared fine-grained meta-category environment. Guided by our theoretical insights, we design a novel Meta-Environment Learner (MEL) for FG-FSCIL, which evolves fine-grained feature extraction to enhance meta-environment understanding and simultaneously regularizes hypothesis space complexity. Extensive experiments demonstrate that our method consistently and significantly outperforms existing approaches.

1 Introduction

Fine-grained classification [12, 33] focuses on distinguishing instances from subordinate categories within the same meta-category (e.g., bird species, car models), and has been widely applied in domains such as ecological protection and security monitoring [28]. Traditionally, research has centered on improving fine-grained feature extraction using fixed and fully annotated datasets. However, in practical scenarios, the collection and annotation of fine-grained data are often limited by domain expertise and time costs, resulting in data scarcity and continuously evolving category sets. To address these challenges, recent studies have explored fine-grained classification under Few-Shot Class-Incremental Learning (FSCIL) [1, 24] setting. FSCIL enables models to learn new classes with limited samples after training on base classes, while retaining the ability to recognize all seen classes. Fine-Grained FSCIL (FG-FSCIL) [18] thus offers a practical and efficient solution for updating models with new fine-grained category information under limited data conditions.

Due to the high discriminative difficulty of fine-grained categories, FG-FSCIL research draws on traditional ideas from fine-grained classification methods and places greater emphasis on extracting fine-grained discriminative features. However, due to the limited training samples available during incremental sessions, this idea still settles for the fine-grained feature extraction capabilities learned

[*] Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

from the base classes. Notably, fine-grained data differs from conventional data in that its sub-categories exhibit smaller inter-class variations, which naturally facilitates synergy across sub-categories. Meanwhile, the incremental learning setting provides the opportunity to progressively enhance this synergy by continuously incorporating new sub-category data over time. Motivated by data characteristics and the learning scenario, we aim to develop a model that not only refines its understanding of the meta-category environment using limited incremental data, but also leverages this evolving understanding to improve classification across all fine-grained sub-categories.

In this paper, we provide a theoretical formulation of the FSCIL problem and derive generalization error bounds for the expected error incurred when learning sub-category samples within a shared meta-environment. While traditional efforts to enhance fine-grained feature extraction help reduce empirical error and thus lower the bound, our analysis reveals that further reducing the bound relies on minimizing the complexity of both per-session sub-categories and the overall meta-environment. Crucially, we show that leveraging incremental sub-categories to enhance understanding of the meta-environment is a breakthrough point. The derived bounds also offer valuable guidance for achieving this under limited training data. Guided by these theoretical insights, we propose a novel Meta-Environment Learner (MEL) for FG-FSCIL, comprising two key processes: Evolving and Regularizing. Specifically, we introduce an incrementally optimized meta-category vector to enable evolving fine-grained feature extraction, enhancing the feature extractor's understanding of the meta-environment while avoiding unnecessary complexity. In parallel, sub-category relationship regularization guides feature space transformation and further constrains hypothesis space complexity, thus further regularizing two complexity terms. Our key contributions are summarized as follows:

- We theoretically formulate the FSCIL problem and analyze its generalization bound within the fine-grained meta-category environment, which is consistent with the core intuitions and motivations.
- Based on the derived generalization bound, we design a novel Meta-Environment Learner (MEL) for FG-FSCIL, which simultaneously evolves fine-grained feature extraction and regularizes hypothesis space complexity.
- Extensive experiments show that our method significantly outperforms both recent FSCIL methods and FG-FSCIL methods, demonstrating the effectiveness of our theoretically grounded design.

2 Related Work

Fine-grained classification [36, 23, 4] focuses on distinguishing visually similar sub-categories within a general meta-category, such as different species of birds, different breeds of dogs, or different models of airplanes. Compared to generic classification tasks, it demands models to capture subtle and localized discriminative features, as inter-class differences are often minimal. As a result, fine-grained classification methods [39, 31, 27] place greater emphasis on learning feature representations than conventional classification approaches. Given the challenges of collecting and annotating fine-grained data in real-world scenarios, Few-Shot Fine-Grained Classification (FSFG) [29, 10, 13, 14, 34] has emerged to address the problem of recognizing novel fine-grained categories using only a few labeled examples. However, existing works overlook the inherent potential of fine-grained sub-categories to collaboratively and incrementally enrich shared meta-category representations, which could serve as a natural foundation for continual learning.

Few-Shot Class-Incremental Learning (FSCIL) [5, 7, 26, 35] is a learning paradigm where the model first learns base classes with sufficient labeled data, then incrementally learns novel classes from only a few labeled samples. To improve performance, most methods [32, 37, 17, 11] leverage pretrained parameters to establish a strong initialization for subsequent learning, or employ data augmentation strategies, including the generation of pseudo-classes to facilitate later sessions. Although [6] includes experiments on fine-grained datasets, it remains a FSCIL method without specific discussion or methodological design for the characteristics of fine-grained data. Compared to traditional datasets, fine-grained datasets exhibit subtler inter-class differences and offer fewer labeled samples in the base session, making it challenging for conventional FSCIL methods to perform effectively in such settings. Recent work [18] has started to explore Fine-Grained Few-Shot Class-Incremental Learning (FG-FSCIL). However, similar to general FSCIL methods, they overlook

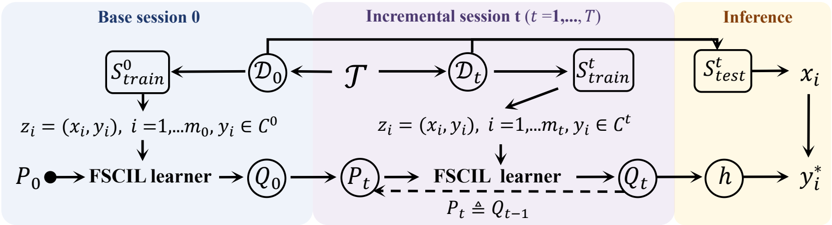


Figure 1: The FSCIL learner with environment $\mathcal{T}$, sample distributions $\mathcal{D}_t$, training set S^t_{train}, prior distribution P_t, posterior distribution Q_t, and a hypothesis h.

the inherent synergy among sub-categories in fine-grained data. As a result, the full potential of incremental learning is not fully exploited.

3 Methodology

In this section, we begin by providing a theoretical formulation of the FSCIL problem. Based on this, we derive a generalization bound, which forms the foundation for designing our FG-FSCIL method.

3.1 Problem Formulation

In FSCIL, the model is trained on a sequence of training sets $\{S^t_{train}\}_{t=0}^{T}$ from session t. Each S^t_{train} comprises a set of pairs $z_i = (x_i, y_i)\,, i = 1, ..., m_t$, where x_i is a sample from class $y_i \in \mathcal{C}^t$. For fine-grained dataset, all classes $\mathcal{C}^0 \cdots \cup \mathcal{C}^T$ belong to the same superclass. $\forall i, j$ and $i \neq j, \mathcal{C}^i \cap \mathcal{C}^j = \varnothing$. The training set S^0_{train} in the base session contains sufficient samples, i.e., m_0 is much larger than others. In each subsequent incremental session ($t \geq 1$), the training set S^t_{train} contains limited samples and follows an N-way K-shot format. This means there are only K samples for each of the N classes, i.e., $m_t = N \times K$ for $t \geq 1$. During the inference stage in session t, the test set S^t_{test} is generated from all sample distributions $\mathcal{D}^0 \cup \mathcal{D}^1 \cdots \cup \mathcal{D}^t$ over all seen classes $\mathcal{C}^0 \cup \mathcal{C}^1 \cdots \cup \mathcal{C}^t$.

To model this setup, following [20, 2, 21], all classes from different sessions are assumed to share the same sample space $\mathcal{Z}$, hypothesis space $\mathcal{H}$, and loss function $\ell : \mathcal{H} \times \mathcal{Z} \to [0, 1]$. As shown in Figure 1, the FSCIL learner can access the dataset S^t_{train} that is sampled i.i.d. from an unknown sample distribution $\mathcal{D}_t$ over the class set $\mathcal{C}^t$ in session t. All sample distributions $\mathcal{D}_t$ are drawn i.i.d from an unknown environment distribution $\mathcal{T}$ [3]. To solve the FSCIL problem, the learner takes as input a training set S^t_{train} and a prior distribution P_t, and output a posterior distribution $Q_t = Q(S^t_{train}, P_t)$ over $\mathcal{H}$, that is, a mapping $\mathcal{Z} \times \mathcal{M} \to \mathcal{M}$, where $\mathcal{M}$ denotes the set of distributions over $\mathcal{H}$. The initial prior P_0 can be modeled as a Gaussian distribution over neural network weights [19]. For subsequent sessions, the prior P_t is set to the posterior from the previous session, i.e. $P_t \triangleq Q_{t-1}$. Then, the expected error and empirical error for each sample distribution D_t in session t' are given by averaging over the posterior distribution $Q_{t'} \in \mathcal{M}$, namely $er\left(Q_{t'}, \mathcal{D}_t\right) \triangleq \mathbb{E}_{h \sim Q_{t'}} \mathbb{E}_{z \sim \mathcal{D}_t} \ell(h, z)$ and $\widehat{er}\left(Q_{t'}, S^t_{train}\right) \triangleq \mathbb{E}_{h \sim Q_{t'}} \left(1/m_t\right) \sum_{i=1}^{m_t} \ell\left(h, z_i\right)$ for a single hypothesis $h \in \mathcal{H}$.

3.2 Generalization Bounds for FSCIL and Meta-Category Environment

Based on the setting introduced in Section 3.1, we derive generalization bounds for FSCIL in session t'. Given the openness of the learning scenario, our goal is to enable the FSCIL learner to extract knowledge from the available categories that can benefit all classes within the same environment.

Therefore, the effectiveness of the posterior distribution $Q_{t'}$ is evaluated by the expected error incurred when learning classes drawn from the same environment $\mathcal{T}$, i.e., $er\left(Q_{t'}, \mathcal{T}\right) \triangleq \mathbb{E}_{\mathcal{D}_t \sim \mathcal{T}} \mathbb{E}_{h \sim Q_{t'}} \mathbb{E}_{z \sim \mathcal{D}_t} \ell(h, z)$. Specifically, we analyze the generalization error for both the newly introduced classes in each session and the overall environment. Formal proofs are provided in Section A.1 of the supplementary material.

Theorem 3.1. $\delta_t \triangleq \frac{\delta}{2(t'+1)\alpha_t}$ *For any $\delta \in (0,1]$, the following bound holds uniformly for posterior distribution $Q_{t'}$, with probability at least $1-\delta$ over a sequence of training set $S^0_{train}, ..., S^{t'}_{train}$,*

$$er(Q_{t'}, \mathcal{T}) \le \sum_{t=0}^{t'} \alpha_t \widehat{er}(Q_{t'}, S^t_{train}) \tag{1}$$

$$+ \sum_{t=0}^{t'} \alpha_t \sqrt{\frac{1}{2(m_t-1)} \left(\sum_{i=t}^{t'} D_{\mathrm{KL}}(Q_i \| P_i) + \log \frac{2(t'+1)\alpha_t m_t}{\delta} \right)} \tag{2}$$

$$+ \sqrt{\frac{1}{2t'} \left(D_{\mathrm{KL}}(Q_{t'} \| P_0) + \log \frac{2(t'+1)}{\delta} \right)}, \tag{3}$$

where $D_{\mathrm{KL}}(\cdot\|\cdot)$ denotes the Kullback–Leibler (KL) divergence and $\alpha_t = |\mathcal{C}^t| / \sum_{i=0}^{t'} |\mathcal{C}^i|$.

The overall bound comprises the **empirical error** (Equation 1) and two complexity terms: the **session complexity** from each sample distribution $\mathcal{D}_t$ (Equation 2) and the **environment complexity** across all sessions (Equation 3), which together offer potent insights for tackling the FSCIL challenge.

- **Insight 0.** *Reducing empirical error $\widehat{er}(Q_{t'}, S^t_{train})$ (Equation 1) can tighten the bound. Since $\alpha_0 > \alpha_t$ ($t \ge 1$), the error on S^0_{train} plays a dominant role in this reduction.* In practice, focusing on fine-grained feature extraction on S^t_{train} can effectively lower $\widehat{er}(Q_{t'}, S^t_{train})$.

Previous FG-FSCIL methods focus on training the feature extractor during session 0 and then freeze it, aligning with **Insight 0** to achieve a smaller overall empirical error. Other common techniques in FSCIL, such as initializing with pretrained models and applying data augmentation, essentially reduce the complexity terms—either by making the initial prior P_0 closer to the final posterior $Q_{t'}$, or by effectively increasing the number of training samples m_t.

A deeper understanding of Theorem 3.1 in the context of fine-grained data enables a more systematic analysis of the inherent synergistic learning potential that is often overlooked by existing FSCIL and FG-FSCIL methods. Based on widely recognized characteristics of fine-grained datasets, we begin by providing the following definition,

Definition 3.2 (Meta-Environment)**.** *In fine-grained settings, where all sub-categories typically belong to the same meta-category, the shared environment $\mathcal{T}$—from which all sample distributions $\mathcal{D}_t$ are drawn—is referred to as the Meta-Environment. This structure implies smaller inter-class variation and facilitates positive transfer across sub-categories.*

Consequently, the FG-FSCIL learner is in a favorable position to leverage the information from seen sub-categories to extract transferable knowledge that benefits all sub-categories within the same meta-category. This makes the environment complexity term a breakthrough point for further reducing the overall generalization error in fine-grained settings:

- **Insight 1.** *As t' increases, the environment complexity (Equation 3) tends to decrease, resulting in a tighter bound.* This suggests that leveraging sub-categories from incremental sessions to train the feature extractor can enhance its understanding of the overall meta-environment—consistent with the intuition and motivation discussed above.
- **Insight 2.** *Since $m_0 \gg m_t (t \ge 1)$, controlling $D_{\mathrm{KL}}(Q_t \| P_t) (t \ge 1)$ can effectively reduce the overall session complexity (Equation 2).* This indicates that while enhancing the feature extractor's understanding of the meta-environment, it is critical to limit significant changes in large parameter subsets during incremental learning.
- **Insight 3.** *Both complexity terms in Equations 2 and 3 characterize data-dependent hypothesis space complexity, which is often positively correlated with the number of learnable parameters.* From a generalization perspective of deep neural networks [38], hypothesis space complexity is primarily governed by classifier parameters, while feature space transformation may potentially simplify the classification.

3.3 Meta-Environment Learner for FSCIL

Building upon the theoretical insights above, we propose a novel Meta-Environment Learner (MEL) for FG-FSCIL. The overall architecture is illustrated in Figure 2. Unlike prior FG-FSCIL methods that

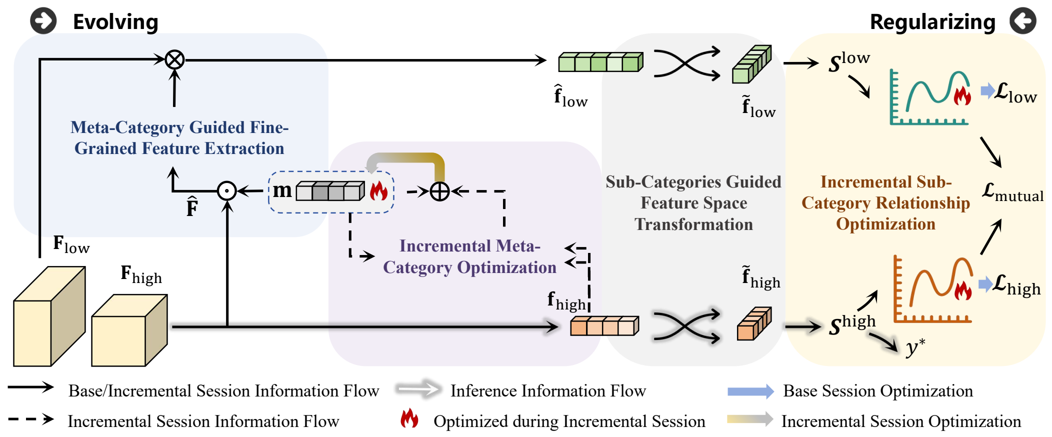

Evolving
Regularizing
Meta-Category Guided Fine-Grained Feature Extraction
$\hat{\mathbf{f}}_{\text{low}}$
$\tilde{\mathbf{f}}_{\text{low}}$
s^{low}
$\mathcal{L}_{\text{low}}$
$\mathbf{m}$
$\hat{\mathbf{F}}$
Incremental Meta-Category Optimization
Sub-Categories Guided Feature Space Transformation
Incremental Sub-Category Relationship Optimization
$\mathcal{L}_{\text{mutual}}$
$\mathbf{F}_{\text{low}}$
$\mathbf{F}_{\text{high}}$
$\mathbf{f}_{\text{high}}$
$\tilde{\mathbf{f}}_{\text{high}}$
s^{high}
$\mathcal{L}_{\text{high}}$
y^*
Base/Incremental Session Information Flow
Inference Information Flow
Base Session Optimization
Incremental Session Information Flow
Optimized during Incremental Session
Incremental Session Optimization

Figure 2: Overall method framework. During the incremental sessions, only Meta-Category Optimization and Sub-category Relationship Optimization are performed. Inference relies solely on high-level features.

freeze the feature extractor during incremental sessions, MEL continuously learns and leverages both the meta-category and its sub-categories. This allows the meta-environment to **evolve** progressively (Section 3.3.1) while remaining properly **regularized** (Section 3.3.2).

3.3.1 Evolving: Incremental Meta-Category Guided Fine-Grained Feature Extraction

Compared to conventional datasets, FG-FSCIL aims to minimize empirical risk (Equation 1) by identifying subtle differences among similar sub-categories through an evolving process (cf. **Insight 0**). To continuously optimize the fine-grained feature extraction process without significantly increasing the complexity terms (Equations 2 and 3), we avoid introducing additional heavy modules and instead primarily leverage the existing backbone structures (cf. **Insight 3**). Accordingly, for any backbone, we denote deeper-layer outputs as high-level semantic features $\mathbf{F}_{\text{high}} \in \mathbb{R}^{C_h \times H_h \times W_h}$, and shallower-layer outputs as low-level fine-grained features $\mathbf{F}_{\text{low}} \in \mathbb{R}^{C_l \times H_l \times W_l}$. Their average pooled representations are denoted as $\mathbf{f}_{\text{high}} \in \mathbb{R}^{C_h}$ and $\mathbf{f}_{\text{low}} \in \mathbb{R}^{C_l}$, respectively.

Meta-Category Guided Fine-Grained Feature Extraction. With a lightweight design, our approach not only utilizes high-level semantic information but also integrates fine-grained details from low-level features. Specifically, to capture instance-specific cues, the high-level semantic features $\mathbf{F}_{\text{high}}$ of each instance can be utilized to spatially localize critical fine-grained features in $\mathbf{F}_{\text{low}}$. To further refine these cues across samples by leveraging the inherent correlations among fine-grained categories, we introduce a learnable meta-category vector $\mathbf{m} \in \mathbb{R}^{C_h}$, which encodes meta-category knowledge in a channel-wise manner with minimal parameter overhead (cf. **Insight 2**),

$$\hat{\mathbf{F}} = \mathbf{F}_{\text{high}} \odot \sigma(\mathbf{m}) \in \mathbb{R}^{C_l \times H_l \times W_l}, \tag{4}$$

where σ denotes the leaky ReLU activation function and $\odot$ indicates element-wise multiplication. To align with the spatial resolution of the low-level feature map $\mathbf{F}_{\text{low}}$, we upsample the cues $\hat{\mathbf{F}}$, and then reduce the channel dimension to generate N attention masks:

$$\mathbf{M} = Sigmoid(g_\phi(\mathcal{I}(\hat{\mathbf{F}}))) \in \mathbb{R}^{N \times H_l \times W_l}, \tag{5}$$

where $\mathcal{I}$ denotes bilinear interpolation, g_ϕ is a 1×1 convolutional block, and $Sigmoid$ denotes the sigmoid activation function. Then, the generated masks are applied to the low-level feature maps to highlight the key regions, forming the final refined fine-grained representation,

$$\hat{\mathbf{f}}_{\text{low}} = Concat(Pool(\mathbf{F}_{\text{low}}, \mathbf{M})) \in \mathbb{R}^{N \cdot C_l}, \tag{6}$$

where $Pool$ denotes masked average pooling under each of the N masks, and $Concat$ represents the concatenation of resulting vectors.

Incremental Meta-Category Optimization. In particular, during incremental sessions, it is essential to utilize newly available data to progressively refine the model's understanding of the meta-environment (cf. **Insight 1**). At the same time, it is important to avoid significant parameter shifts—especially in scenarios with limited training data (cf. **Insight 2**).

To this end, based on Equation 4, the fine-grained feature learning process can be continuously optimized by updating the meta-category vector $\mathbf{m}$ using a limited number of training samples. During the incremental session t', we compute the relation coefficients between the sample semantic representations $\mathbf{f}^i_{\text{high}}$ and the existing meta-category vector $\mathbf{m}$,

$$a_i = \mathbf{m}^\top \mathbf{f}^i_{\text{high}} / \sqrt{C_h} \in \mathbb{R}, i = 1, ..., m_t, \tag{7}$$

The final attention weights corresponding to the meta-category vector are obtained via a softmax function,

$$\hat{a}_i = \frac{\exp(a_i)}{\sum_{j=1}^{m_t} \exp(a_j)}. \tag{8}$$

Then, the meta-category vector can absorb semantic information from all training samples based on the attention weights and update itself accordingly,

$$\mathbf{m} \leftarrow (1-\alpha)\mathbf{m} + \alpha \textstyle\sum_{i=1}^{m_t} (\hat{a}_i \mathbf{f}^i_{\text{high}}) \in \mathbb{R}^{C_h}, \alpha = |\mathcal{C}^{t'}| / \textstyle\sum_{t=0}^{t'} |\mathcal{C}^t|. \tag{9}$$

where α is a class proportion-based weight. This attention-based aggregation over new training samples further constrains the update magnitude of the meta-category vector, thereby preventing a substantial increase in the KL divergence.

3.3.2 Regularizing: Incremental Sub-categories Guided Feature Space Transformation

Considering the hypothesis space complexity of deep neural networks, the two complexity terms (Equations 2 and 3) can be further regularized before the final classification step (cf. **Insight 3**). On the one hand, with proper guidance, feature space transformation enables the simplification of fine-grained classification boundaries, making them easier to model with simpler classifier parameters. On the other hand, consistency regularization between fine-grained and semantic representations provides directional guidance for the transformation, encouraging a reduction in mutual information between input and representation. This helps eliminate redundant information and constrains the hypothesis space to focus on task-relevant variations.

Sub-categories Guided Feature Space Transformation. To retain complementary information from both the original and transformed spaces, the two type representations $\hat{\mathbf{f}}_{\text{low}}$ and $\mathbf{f}_{\text{high}}$ obtained in Section 3.3.1 are first individually mapped into a new feature space, and then fused with their original counterparts,

$$\tilde{\mathbf{f}}_{\text{low}} = (\hat{\mathbf{f}}_{\text{low}} + h_\phi(\hat{\mathbf{f}}_{\text{low}}))/2, \quad \tilde{\mathbf{f}}_{\text{high}} = (\mathbf{f}_{\text{high}} + h_\phi(\mathbf{f}_{\text{high}}))/2, \tag{10}$$

where h_ϕ denotes a fully connected block. During the base training phase, the overall loss function guided by sub-categories for feature extraction and feature space transformation is defined as,

$$\mathcal{L} = (1-\beta)\mathcal{L}_{\text{high}} + \beta\mathcal{L}_{\text{low}} + \gamma\mathcal{L}_{\text{mutual}}, \tag{11}$$

where β and γ are hyperparameters that control the weights of each term, and $\mathcal{L}_{\text{mutual}}$ denotes the consistency regularization term. Both $\mathcal{L}_{\text{high}}$ and $\mathcal{L}_{\text{low}}$ are cross-entropy losses,

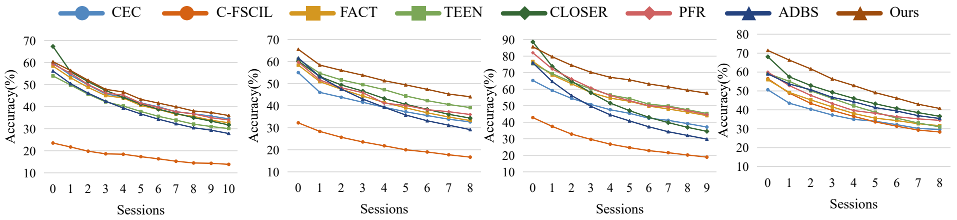
$$\mathcal{L}_* = \frac{1}{m_0} \sum_{i=1}^{m_0} -\log \Phi(\tau \mathcal{S}^*_{i,1:|\mathcal{C}^{(0)}|}), * \in \{\text{high}, \text{low}\}, \tag{12}$$

where $\mathcal{S}^*_{i,j}$ denotes cosine similarity between $\tilde{\mathbf{f}}^i_*$ and classifier $\mathbf{w}^j_*$, τ is the temperature parameter, and Φ is defined as the softmax operation,

$$\Phi(\tau \mathcal{S}^*_{i,1:|\mathcal{C}^{(0)}|}) = \frac{\exp\left(\tau \mathcal{S}^*_{i,y_i}\right)}{\sum_{j=1}^{|\mathcal{C}^{(0)}|} \exp\left(\tau \mathcal{S}^*_{i,j}\right)}. \tag{13}$$

Table 1: Performance on four fine-grained benchmark datasets. Due to space limitations, only the accuracy of the final session (Acc$_T$) and average accuracy across all sessions (Acc$_{Avg}$) are presented.

Method	CUB200		Stanford Dogs		Stanford Cars		FGVCAircraft	
	Acc$_T$	Acc$_{Avg}$	Acc$_T$	Acc$_{Avg}$	Acc$_T$	Acc$_{Avg}$	Acc$_T$	Acc$_{Avg}$
CEC [32]	34.50	43.71	32.65	40.56	37.09	48.24	29.57	36.94
C-FSCIL [7]	13.86	17.62	16.68	22.78	18.90	27.76	28.28	38.60
FACT [37]	32.71	42.50	33.26	42.96	43.98	56.17	31.60	40.46
TEEN [26]	30.08	39.33	39.14	47.86	45.32	57.37	30.98	43.42
CLOSER [17]	31.74	43.95	34.37	44.69	34.49	53.78	36.56	48.08
PFR [18]	33.99	43.70	36.01	44.25	44.56	58.12	34.42	43.05
ADBS [11]	27.85	38.71	29.15	41.54	29.79	46.50	35.42	45.23
Ours	**36.00**	**45.44**	**44.06**	**52.38**	**57.58**	**68.43**	**40.72**	**54.17**



(a) CUB200 (b) Stanford Dogs (c) Stanford Cars (d) FGVCAircraft

Figure 3: Classification accuracy for each session on four fine-grained benchmark datasets.

Incremental Sub-category Relationship Optimization. Meanwhile, $\mathcal{L}_{\text{mutual}}$ is used to continuously constrain the sub-category relationship distributions between the fine-grained and semantic representations,

$$\mathcal{L}_{\text{mutual}} = \frac{1}{m_0} \sum_{i=1}^{m_0} \Big[D_{\text{KL}} \left(\Phi(\tau \mathcal{S}^{\text{low}}_{i,1:|\mathcal{C}^{(0)}|}) \parallel \Phi(\tau \mathcal{S}^{\text{high}}_{i,1:|\mathcal{C}^{(0)}|}) \right) + D_{\text{KL}} \left(\Phi(\tau \mathcal{S}^{\text{high}}_{i,1:|\mathcal{C}^{(0)}|}) \parallel \Phi(\tau \mathcal{S}^{\text{low}}_{i,1:|\mathcal{C}^{(0)}|}) \right) \Big] \tag{14}$$

where $D_{\text{KL}}(p \parallel q) = \sum_j p_j \log p_j / q_j$. After the base training phase, as in most FSCIL methods, the classifier is replaced with prototype-based classifiers and can be extended during the incremental session to accommodate novel classes, $\mathbf{w}^c_* = \frac{1}{Num_c} \sum_{y_i=c} \tilde{\mathbf{f}}^i_*$, where Num_c denotes the number of training samples for class c. Therefore, during the incremental training phase, the mutual learning regularization term $\mathcal{L}_{\text{mutual}}$ can be computed on the new training data to maintain and refine the novel sub-category relationship distributions, enabling ongoing interaction and refinement between the two types of sub-category prototype representations. This regularization not only constrains the feature space used for classification but also encourages mutual complementarity between the two types of representations, thereby implicitly integrating fine-grained knowledge into the semantic representations.

Thus, the classification inference in session t' can be performed using only $\tilde{\mathbf{f}}_{\text{high}}$,

$$y_i^\star = \underset{c \in \bigcup_{t=0}^{t'} \mathcal{C}^t}{\text{argmax}} \, \mathcal{S}^{\text{high}}_{i,1:\sum_{t=0}^{t'} |\mathcal{C}^{(t)}|}. \tag{15}$$

4 Experiments

4.1 Experimental Setup

Datasets. Following the benchmark setting in [18], we evaluate our method on four fine-grained datasets: CUB200 [25], Stanford Dogs [8], Stanford Cars [30], and FGVCAircraft [15]. The data split details are provided in Appendix A.3.

Table 2: Ablation studies of our proposed method on the Stanford Dogs dataset.

Model	Infer	Accuracy in each session (%)								
		0	1	2	3	4	5	6	7	8
M_0	$\mathcal{S}^{\text{high}}$	62.78	55.10	51.55	48.68	45.25	42.15	40.12	38.23	36.68
M_1	$\mathcal{S}^{\text{high}}\mathcal{S}^{\text{low}}$	65.58	57.32	53.82	51.08	48.07	45.93	43.45	41.45	39.82
M_2	$\mathcal{S}^{\text{high}}\mathcal{S}^{\text{low}}$	**66.51**	58.37	54.70	52.05	49.08	46.79	44.44	42.28	40.63
M_3	$\mathcal{S}^{\text{high}}\mathcal{S}^{\text{low}}$	**66.51**	58.40	55.12	52.59	49.73	47.70	45.37	43.22	41.50
M_4	$\mathcal{S}^{\text{high}}\mathcal{S}^{\text{low}}$	66.16	58.27	54.63	52.06	49.67	47.40	45.05	43.28	41.60
M_5	$\mathcal{S}^{\text{high}}\mathcal{S}^{\text{low}}$	65.78	**58.44**	**56.03**	53.48	50.87	48.95	46.97	44.68	43.36
M_5	$\mathcal{S}^{\text{low}}$	65.03	57.68	55.40	52.72	49.97	48.03	45.98	43.67	42.30
M_5	$\mathcal{S}^{\text{high}}$	65.57	**58.44**	**56.03**	**53.84**	**51.34**	**49.45**	**47.42**	**45.29**	**44.06**

Experimental details. Following [18], we employ ResNet-12 as the backbone. In our experiments, $\mathbf{F}_{\text{high}}$ is the output of the final layer of the backbone, while $\mathbf{F}_{\text{low}}$ is the output of the penultimate layer. Following previous works [18], the temperature hyperparameter τ is set to 16. N is set as an integer 4 to ensure diversity in fine-grained features and kept small for efficiency. β and γ are set to 0.5 to balance the optimization strength among different components. For comparative methods whose results are not reported in [18], we reproduce their performance under the same experimental settings using the publicly available source code. Please see Appendix A.3 for details.

4.2 Comparison Results

In this section, we compare our method with recent FSCIL methods and FG-FSCIL method in Table 1 and Figure 3. It can be observed that our method achieves the best overall performance. Specifically, our method achieves a 1.21% higher average accuracy (Acc$_{Avg}$) than CLOSER on the CUB200 dataset, and notably outperforms it by 4.26% in the final session (Acc$_T$). This suggests that our method, by progressively evolving fine-grained feature extraction while constraining hypothesis space complexity, effectively improves performance on incremental classes with limited samples, often leading to a more pronounced gain in the final session. Although ADBS considers model adaptation during the incremental sessions, it lacks mechanisms for inter-class synergy modeling and hypothesis space complexity regularization, resulting in limited performance. Detailed experimental results can be found in Appendix A.4.

4.3 Ablation Study

To demonstrate the effectiveness of each component in our method, we conduct ablation experiments in Table 2. $\mathcal{S}^{\text{high}}$ denotes using only high-level features during inference, as described in Equation 15, while $\mathcal{S}^{\text{low}}$ denotes using only low-level features. $\mathcal{S}^{\text{high}}\mathcal{S}^{\text{low}}$ represents using the sum of their similarity scores. M_0 represents the baseline.

(1) Firstly, M_1 applies fine-grained feature extraction without the meta-category vector (i.e., under Equations 5-6, $\hat{\mathbf{F}}$ is directly set to $\mathbf{F}_{\text{high}}$), together with the basic cross-entropy loss (Equation 12), yields an initial performance improvement. (2) M_2 builds on M_1 by introducing the meta-category vector (Equation 4), (3) and M_3 further performs incremental optimization of this meta-category vector (Equations 7-9), leading to additional performance gains. Further analysis in Section 4.4.1 demonstrates the effectiveness of our optimization strategy. The improved fine-grained feature extraction benefits all sub-categories within the meta-category environment, yielding relatively consistent performance improvements across sessions. (4) M_4 adds feature space transformation (Equation 10) to M_3 does not lead to significant performance gains. (5) However, M_5, which further includes sub-category relationship regularization (Equation 14), shows clear improvement. This indicates that regularizing the alignment between the two relationship distributions enables the feature space transformation process to eliminate redundancy while preserving classification-relevant information, effectively reducing hypothesis space complexity and, consequently, the two complexity terms. This reduction is especially beneficial for classes with fewer training samples, leading to more

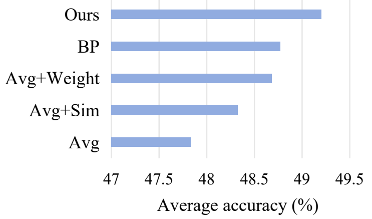
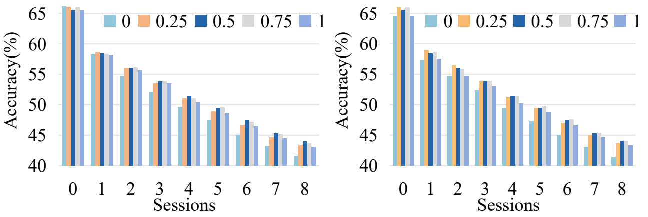


Figure 4: Analysis of meta-category optimization strategies.



(a) Effect of γ ($\beta = 0.5$)

(b) Effect of β ($\gamma = 0.5$)

Figure 5: The influence of hyperparameters γ and β.

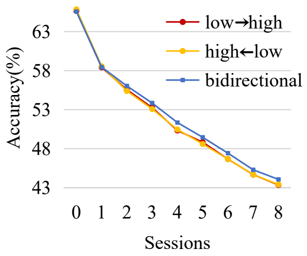
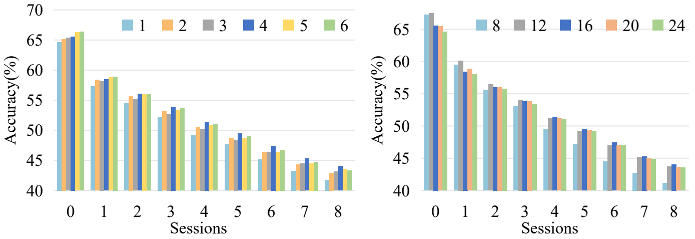


Figure 6: Analysis of $\mathcal{L}_{\text{mutual}}$.



(a) Effect of N

(b) Effect of τ

Figure 7: The influence of hyperparameters N and τ.

noticeable performance improvements in the later incremental sessions. (6) Besides, regularization allows inference using only high-level features ($\mathcal{S}^{\text{high}}$) to achieve performance comparable to or better than using both ($\mathcal{S}^{\text{high}}\mathcal{S}^{\text{low}}$). Further analysis in Section 4.4.2 demonstrates the impact of fine-grained feature extraction and regularization on class representations.

4.4 Further Analysis

4.4.1 Different Meta-Category Optimization Strategies

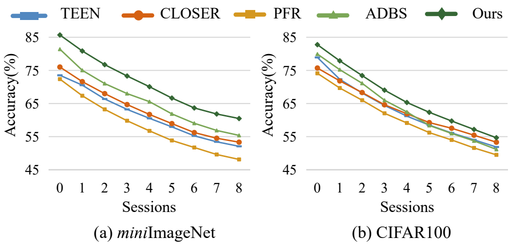
To demonstrate the effectiveness of our meta-category optimization strategy (Equations 7-9 in Section 3.3.1), we compare the average accuracy across incremental sessions on the Stanford Dogs dataset under different optimization schemes, as shown in Figure 4.

It can be observed that directly updating the meta-category vector $\mathbf{m}$ by averaging it with the feature vectors $\mathbf{f}^i_{\text{high}}$ of all training samples (**Avg**) leads to poor performance. Introducing cosine similarity as a soft weight between $\mathbf{m}$ and the mean of $\mathbf{f}^i_{\text{high}}$ (**Avg+Sim**) slightly limits the update magnitude but still performs poorly. This suggests that even with few parameters, optimization under limited samples can easily cause posterior drift — a key reason why existing FG-FSCIL methods do not incorporate new class knowledge into the feature extractor. Replacing cosine similarity with a class proportion-based weight α (**Avg+Weight**) helps reduce the update magnitude in later sessions, leading to moderate improvements. Our proposed method (**Ours**, Equation 9) first computes a cosine-weighted sum of all $\mathbf{f}^i_{\text{high}}$ to mitigate the influence of outliers, then applies α to further constrain the impact of new classes—yielding the best performance. While backpropagation-based optimization (**BP**) also perform well, they require careful tuning of learning rate and training epochs. In contrast, our strategy achieves effective updates with no additional hyperparameters.

4.4.2 Effect of Terms $\mathcal{L}_{\text{mutual}}$ and $\mathcal{L}_{\text{low}}$

To analyze the effects of the regularization term $\mathcal{L}_{\text{mutual}}$ and the fine-grained feature extraction term $\mathcal{L}_{\text{low}}$ (Section 3.3.2), we evaluate the influence of their corresponding hyperparameters γ and β (Equation 11) on the Stanford Dogs dataset, as shown in Figure 5.

Figure 5(a) shows that in session 0, the value of γ (i.e., whether the regularization term is applied) has little effect on performance. This suggests that for base classes with ample training samples (large m_0), high-level semantic features already generalize well, and regularization has limited impact on the complexity term (Equation 2). However, as subsequent sessions introduce more new



(a) *mini*ImageNet (b) CIFAR100

Figure 8: Classification accuracy on conventional FSCIL benchmarks.

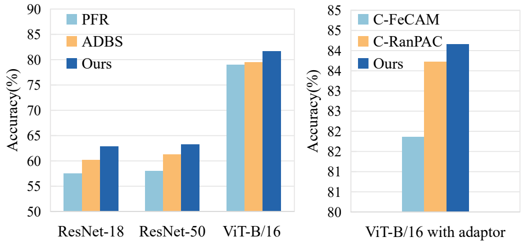


Figure 9: Accuracy after the last session on CUB200 with different backbones.

classes with limited training samples, the advantage of regularization gradually emerges, consistently outperforming the no-regularization setting across different values. Figure 5(b) shows that applying fine-grained feature extraction ($\beta > 0$) consistently improves performance across all sessions. This is particularly evident in later sessions with more incremental classes, where limited training samples lead to less reliable semantic representations, making fine-grained features more critical for recognition—i.e., a larger β is needed. Optimal performance is generally achieved when γ and β are set around 0.5. Furthermore, we analyze the difference between using bidirectional KL divergence (i.e., $\mathcal{L}_{\text{mutual}}$) and unidirectional KL divergence (i.e., only high $\rightarrow$ low or only low $\rightarrow$ high). As shown in Figure 6, bidirectional KL divergence leads to better performance. Our approach adopts bidirectional KL divergence to enable mutual regularization and refinement between the two branches, thereby achieving better alignment of their sub-category relationship distributions.

4.4.3 Effect of Hyperparameters N and τ

In this section, we analyze the effect of other hyperparameters, the number of attention masks N and the temperature τ. The experimental results are shown in Figure 7.

It can be observed that MEL's performance remains relatively stable across a range of values for these hyperparameters. When $N \geq 4$, increasing N yields no additional performance gain, with $N = 4$ achieving the best overall performance, particularly in the later sessions. Additionally, consistent with other FSCIL methods, our MEL demonstrates sustained performance advantages when $\tau = 16$. These results are consistent with the principles behind our chosen hyperparameter settings.

4.4.4 Performance on Other FSCIL Benchmarks and Backbones

Although the analysis primarily focuses on fine-grained datasets and the proposed method is tailored for FG-FSCIL, it still outperforms recent methods on conventional FSCIL benchmarks [22, 9], as shown in Figure 8. This is because the generalization error bound analysis for FSCIL remains valid. While fine-grained feature extraction is not crucial for conventional datasets, constraining hypothesis space complexity during empirical error minimization is still essential. Moreover, our method still achieves better performance than other approaches (e.g., PFR [18], ADBS [11], and C-FeCAM/C-RanPAC [6]) across different backbones, as shown in Figure 9.

5 Conclusion

In conclusion, we revisit the Fine-Grained Few-Shot Class-Incremental Learning (FG-FSCIL) problem from a theoretical perspective and formulate it within a shared meta-environment. By deriving a generalization error bound, we demonstrate the importance of further minimizing the complexity of both per-session sub-categories and the overall meta-environment. Building upon these insights, we propose the Meta-Environment Learner (MEL), which progressively evolves fine-grained feature representations while regularizing hypothesis space complexity. Extensive experiments on benchmark datasets validate the effectiveness of our approach, showing consistent and significant improvements over state-of-the-art methods.

Acknowledgments and Disclosure of Funding

This work was supported in part by the National Natural Science Foundation of China under Grant 62202272, 62172256, 62202278, 62302276, in part by the Natural Science Foundation of Shandong Province under Grant ZR2024LZH002 and Taishan Scholar Project of Shandong Province under Grant tstp20250704.

References

[1] Idan Achituve, Aviv Navon, Yochai Yemini, Gal Chechik, and Ethan Fetaya. Gp-tree: A gaussian process classifier for few-shot incremental learning. In *Int. Conf. on Mach. Learn.*, pages 54–65, 2021.

[2] Ron Amit and Ron Meir. Meta-learning by adjusting priors based on extended pac-bayes theory. In *Int. Conf. on Mach. Learn.*, pages 205–214, 2018.

[3] Jonathan Baxter. A model of inductive bias learning. *J. Artif. Intell. Res.*, 12:149–198, 2000.

[4] Yue Chen, Yalong Bai, Wei Zhang, and Tao Mei. Destruction and construction learning for fine-grained image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5157–5166, 2019.

[5] Songlin Dong, Xiaopeng Hong, Xiaoyu Tao, Xinyuan Chang, Xing Wei, and Yihong Gong. Few-shot class-incremental learning via relation knowledge distillation. In *Proc. AAAI Conf. Artif. Intell.*, pages 1255–1263, 2021.

[6] Dipam Goswami, Bartlomiej Twardowski, and Joost van de Weijer. Calibrating higher-order statistics for few-shot class-incremental learning with pre-trained vision transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4075–4084, 2024.

[7] Michael Hersche, Geethan Karunaratne, Giovanni Cherubini, Luca Benini, Abu Sebastian, and Abbas Rahimi. Constrained few-shot class-incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9047–9057, 2022.

[8] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization : Stanford dogs. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2012.

[9] Alex Krizhevsky. Learning multiple layers of features from tiny images. *Handbook of Systemic Autoimmune Diseases*, 1(4), 2009.

[10] Su Been Lee, WonJun Moon, and Jae-Pil Heo. Task discrepancy maximization for fine-grained few-shot classification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5321–5330, 2022.

[11] Linhao Li, Yongzhang Tan, Siyuan Yang, Hao Cheng, Yongfeng Dong, and Liang Yang. Adaptive decision boundary for few-shot class-incremental learning. In *Proc. AAAI Conf. Artif. Intell.*, pages 18359–18367, 2025.

[12] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear CNN models for fine-grained visual recognition. In *Int. Conf. Comput. Vis.*, pages 1449–1457, 2015.

[13] Zhen-Xiang Ma, Zhen-Duo Chen, Li-Jun Zhao, Zi-Chao Zhang, Xin Luo, and Xin-Shun Xu. Cross-layer and cross-sample feature optimization network for few-shot fine-grained image classification. In *Proc. AAAI Conf. Artif. Intell.*, pages 4136–4144, 2024.

[14] Zhen-Xiang Ma, Zhen-Duo Chen, Li-Jun Zhao, Zi-Chao Zhang, Tai Zheng, Xin Luo, and Xin-Shun Xu. Bi-directional task-guided network for few-shot fine-grained image classification. In *ACM Int. Conf. Multimedia*, pages 8277–8286, 2024.

[15] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew B. Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *CoRR*, abs/1306.5151, 2013.

[16] David A. McAllester. Pac-bayesian model averaging. In *COLT*, pages 164–170, 1999.

[17] Junghun Oh, Sungyong Baik, and Kyoung Mu Lee. CLOSER: towards better representation learning for few-shot class-incremental learning. In *Eur. Conf. Comput. Vis.*, pages 18–35, 2024.

[18] Zicheng Pan, Weichuan Zhang, Xiaohan Yu, Miaohua Zhang, and Yongsheng Gao. Pseudo-set frequency refinement architecture for fine-grained few-shot class-incremental learning. *Pattern Recognition*, 155:110686, 2024.

[19] Anastasia Pentina and Christoph H. Lampert. A pac-bayesian bound for lifelong learning. In *Int. Conf. on Mach. Learn.*, pages 991–999, 2014.

[20] Anastasia Pentina and Christoph H. Lampert. Lifelong learning with non-i.i.d. tasks. In *Adv. Neural Inform. Process. Syst.*, pages 1540–1548, 2015.

[21] Jonas Rothfuss, Martin Josifoski, Vincent Fortuin, and Andreas Krause. Scalable pac-bayesian meta-learning via the pac-optimal hyper-posterior: From theory to practice. *J. Mach. Learn. Res.*, 24:386:1–386:62, 2023.

[22] 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 Li Fei-Fei. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015.

[23] Ming Sun, Yuchen Yuan, Feng Zhou, and Errui Ding. Multi-attention multi-class constraint for fine-grained image recognition. In *Eur. Conf. Comput. Vis.*, pages 834–850, 2018.

[24] Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12180–12189, 2020.

[25] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge J. Belongie. The caltech-ucsd birds-200-2011 dataset. *California Inst. Technol.*, 2011.

[26] Qi-Wei Wang, Da-Wei Zhou, Yi-Kai Zhang, De-Chuan Zhan, and Han-Jia Ye. Few-shot class-incremental learning via training-free prototype calibration. In *Adv. Neural Inform. Process. Syst.*, 2023.

[27] Shijie Wang, Zhihui Wang, Haojie Li, Jianlong Chang, Wanli Ouyang, and Qi Tian. Semantic-guided information alignment network for fine-grained image recognition. *IEEE Trans. Circuit Syst. Video Technol.*, 33(11):6558–6570, 2023.

[28] Xiu-Shen Wei, Yi-Zhe Song, Oisin Mac Aodha, Jianxin Wu, Yuxin Peng, Jinhui Tang, Jian Yang, and Serge J. Belongie. Fine-grained image analysis with deep learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(12):8927–8948, 2022.

[29] Xiu-Shen Wei, Peng Wang, Lingqiao Liu, Chunhua Shen, and Jianxin Wu. Piecewise classifier mappings: Learning fine-grained learners for novel categories with few examples. *IEEE Trans. Image Process.*, 28(12):6116–6125, 2019.

[30] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3973–3981, 2015.

[31] Xuhui Yang, Yaowei Wang, Ke Chen, Yong Xu, and Yonghong Tian. Fine-grained object classification via self-supervised pose alignment. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7389–7398, 2022.

[32] Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12455–12464, 2021.

[33] Xiaopeng Zhang, Hongkai Xiong, Wengang Zhou, Weiyao Lin, and Qi Tian. Picking deep filter responses for fine-grained image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1134–1142, 2016.

[34] Li-Jun Zhao, Zhen-Duo Chen, Zhen-Xiang Ma, Xin Luo, and Xin-Shun Xu. Angular isotonic loss guided multi-layer integration for few-shot fine-grained image classification. *IEEE Trans. Image Process.*, 33:3778–3792, 2024.

[35] Li-Jun Zhao, Zhen-Duo Chen, Yongxin Wang, Xin Luo, and Xin-Shun Xu. Attraction diminishing and distributing for few-shot class-incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 25657–25666, 2025.

[36] Heliang Zheng, Jianlong Fu, Tao Mei, and Jiebo Luo. Learning multi-attention convolutional neural network for fine-grained image recognition. In *Int. Conf. Comput. Vis.*, pages 5219–5227, 2017.

[37] Da-Wei Zhou, Fu-Yun Wang, Han-Jia Ye, Liang Ma, Shiliang Pu, and De-Chuan Zhan. Forward compatible few-shot class-incremental learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9036–9046, 2022.

[38] Zhi-Hua Zhou. Why over-parameterization of deep neural networks does not overfit? *Sci. China Inf. Sci.*, 64(1), 2021.

[39] Peiqin Zhuang, Yali Wang, and Yu Qiao. Learning attentive pairwise interaction for fine-grained classification. In *Proc. AAAI Conf. Artif. Intell.*, pages 13130–13137, 2020.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See the abstract and last paragraph in Section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Appendix A.2.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: See Appendix A.1.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Section 4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The source code is provided in the supplemental materials. It includes instructions on data access and preparation, and scripts.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Section 4.1 and Appendix A.3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: In the field of Few-Shot Class-Incremental Learning and Fine-Grained Few-Shot Class-Incremental Learning, publicly available codes such as CEC and PFR specify the random seed as 1. Therefore, we follow the same random seed to conduct experiments and compare with other methods without the error bar.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Appendix A.3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have read the guidelines and ensured that our paper conforms to them.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Our method is an academic research for Few-Shot Class-Incremental Learning (FSCIL) and does not pertain to any potential societal impacts on privacy, public health, fairness, or other societal issues.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited [18, 25, 8, 30, 15, 22, 9]. See Section 4.1 and Appendix A.3 for details.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

A Technical Appendices and Supplementary Material

A.1 Proof of the Generalization Bound for FSCIL

In this section, we prove Theorem 3.1 utilizing classical PAC-Bayes bound.

Theorem A.1 (PAC-Bayes Bound [16])**.** *Let $\mathcal{X}$ be a sample space and $\mathbb{X}$ a distribution over $\mathcal{X}$, and let $\mathcal{F}$ be a hypothesis class of functions on $\mathcal{X}$. Let $g : \mathcal{F} \times \mathcal{X} \to [0, 1]$ be a loss function. Suppose $X_1, \ldots, X_K$ are i.i.d. samples from $\mathbb{X}$ and let π be a prior distribution over $\mathcal{F}$ (independent of the samples). Then for any $\delta \in (0, 1]$, the following holds uniformly for all posterior distributions ρ over $\mathcal{F}$ (even sample dependent) with probability at least $1 - \delta$:*

$$\mathbb{E}_{f\sim\rho}\mathbb{E}_{X\sim\mathbb{X}}[g(f, X)] \leq \frac{1}{K}\sum_{k=1}^{K}\mathbb{E}_{f\sim\rho}[g(f, X_k)] + \sqrt{\frac{1}{2(K-1)}\left(D_{\mathrm{KL}}(\rho\|\pi) + \log\frac{K}{\delta}\right)}. \tag{16}$$

where $D_{\mathrm{KL}}(\rho\|\pi)$ denotes the Kullback–Leibler (KL) divergence between the posterior distribution ρ and the prior distribution π, $D_{\mathrm{KL}}(Q\|P) \triangleq \mathop{\mathbb{E}}_{h\sim Q} \log \frac{Q(h)}{P(h)}$.

Proof. According to Theorem A.1, we analyze the generalization error for the classes introduced in each session. The samples are $X_K \triangleq z_{t,i}, K \triangleq m_t$, and their distribution is $\mathbb{X} \triangleq \mathcal{D}_t$. Formally, we set $\rho = Q_{t'}$ and $\pi = P_t$. We get that for any $\delta_t > 0$, with probability at least $1 - \delta_t$ over $S^t_{train} \sim \mathcal{D}_t^{m_t}$,

$$er(Q_{t'}, \mathcal{D}_t) \leq \widehat{er}(Q_{t'}, S^t_{train}) + \sqrt{\frac{1}{2(m_t-1)}\left(D_{\mathrm{KL}}(Q_{t'}\|P_t) + \log\frac{m_t}{\delta_t}\right)}, \tag{17}$$

where $P_t \triangleq Q_{t-1}$ and $Q_{t'}$ represents the posterior distribution in session t', i.e., $Q_{t'} \triangleq Q(S_{t'}, P_{t'})$.

Then, we consider the generalization across $\mathcal{D}_t^{m_t}$ from different sessions, thereby providing a bound on the environment-level generalization. Let $(\mathcal{D}_t, m_t, S^t_{train})$ for $t = 0, 1, \ldots, t'$ be drawn i.i.d. from the environment $\mathcal{T}$ and $S^t_{train} \sim \mathcal{D}_t^{m_t}$. Using Theorem A.1 again, let the prior be $\pi = P_0$ and posterior $\rho = Q_{t'}$. Then, for any $\delta_0 > 0$, with probability at least $1 - \delta_0$,

$$er(Q_{t'}, \mathcal{T}) \leq \sum_{t=0}^{t'}\alpha_t er(Q_{t'}, \mathcal{D}_t) + \sqrt{\frac{1}{2t'}\left(D_{\mathrm{KL}}(Q_{t'}\|P_0) + \log\frac{t'+1}{\delta_0}\right)}, \tag{18}$$

where $\alpha_t = \frac{|\mathcal{C}^t|}{\sum_{i=0}^{t'}|\mathcal{C}^i|}$, reflecting the relationship among the class sets introduced in different sessions.

Finally, to bound the probability of the intersection of the events defined in Equations 17 and 18, we make use of the union bound. Let $\delta > 0$, set $\delta_0 \triangleq \frac{\delta}{2}$ and $\delta_t \triangleq \frac{\delta}{2(t'+1)\alpha_t}$ for $t = 0, 1, ..., t'$. By applying the union bound argument (Lemma A.2), with probability at least $1 - \delta$,

$$er(Q_{t'}, \mathcal{T}) \leq \sum_{t=0}^{t'}\alpha_t\widehat{er}(Q_{t'}, S^t_{train}) \tag{19}$$

$$+ \sum_{t=0}^{t'}\alpha_t\sqrt{\frac{1}{2(m_t-1)}\left(D_{\mathrm{KL}}(Q_{t'}\|P_t) + \log\frac{m_t}{\delta_t}\right)} \tag{20}$$

$$+ \sqrt{\frac{1}{2t'}\left(D_{\mathrm{KL}}(Q_{t'}\|P_0) + \log\frac{t'+1}{\delta_0}\right)}, \tag{21}$$

where $D_{\mathrm{KL}}(Q_{t'}\|P_t) = \sum_{i=t}^{t'} D_{\mathrm{KL}}(Q_i\|P_i)$. The probability is taken over sampling of $(\mathcal{D}_t, m_t) \sim \mathcal{T}$ and $S^t_{train} \sim \mathcal{D}_t^{m_t}, t = 0, 1, ..., t'$. □

In the following, we present the statement of Lemma A.2 along with its proof.

Lemma A.2. *Let $\{E_i\}_{i=1}^n$ be a set of events such that $\mathbb{P}(E_i) \geq 1 - \delta_i$, for some $\delta_i \geq 0$, $i = 1, \dots, n$. Then,*

$$\mathbb{P}\left(\bigcap_{i=1}^n E_i\right) \geq 1 - \sum_{i=1}^n \delta_i.$$

Proof. We begin by observing that

$$\mathbb{P}\left(\bigcap_{i=1}^n E_i\right) = 1 - \mathbb{P}\left(\bigcup_{i=1}^n E_i^c\right), \tag{22}$$

where E_i^c denotes the complement of event E_i.

By applying the union bound, we obtain

$$\mathbb{P}\left(\bigcup_{i=1}^n E_i^c\right) \leq \sum_{i=1}^n \mathbb{P}(E_i^c) = \sum_{i=1}^n (1 - \mathbb{P}(E_i)). \tag{23}$$

Using the assumption $\mathbb{P}(E_i) \geq 1 - \delta_i$, we have

$$\sum_{i=1}^n (1 - \mathbb{P}(E_i)) \leq \sum_{i=1}^n \delta_i. \tag{24}$$

Combining the above results gives

$$\mathbb{P}\left(\bigcap_{i=1}^n E_i\right) \geq 1 - \sum_{i=1}^n \delta_i. \tag{25}$$

□

A.2 Limitations and Future Work

Following the conventional FG-FSCIL setting, our method assumes base and novel classes are sampled from the same domain. This is a standard and practical choice in prior studies, ensuring consistency in semantic granularity. Nevertheless, real-world scenarios may involve more diverse data sources, such as photographs and sketches. Extending the framework to cross-domain FG-FSCIL is a natural and worthwhile extension, which we plan to explore in future work.

A.3 Detailed Experimental Setups

Datasets. For CUB200, 100 classes are used for base training, and the remaining 100 classes are split into 10 incremental sessions. For Stanford Dogs, 80 classes are used in the base session, and the remaining 40 classes are evenly divided into 8 incremental sessions. Each incremental class contains 5 samples, forming a 5-way 5-shot FSCIL setting. For Stanford Cars, 106 classes are used for base training, and the remaining 90 classes are split across 9 sessions, following a 10-way 5-shot setting. For FGVCAircraft, 60 classes are used for base training and 40 for incremental learning. A 5-way 5-shot setting is applied, resulting in 9 sessions in total.

Experimental details. Our implementation is based on the code released by PFR [18] under the MIT license. Our method is conducted with PyTorch library on a single NVIDIA 3090, and SGD with momentum is used for optimization. All method-agnostic hyperparameter settings are kept consistent with those of PFR.

A.4 Detailed Experimental Results and Analysis

Table 3: Classification accuracy on the CUB200 dataset.

Method	Accuracy in each session (%)										
	0	1	2	3	4	5	6	7	8	9	10
CEC	59.94	54.31	49.88	45.81	44.44	41.72	39.95	37.74	36.79	35.74	34.50
C-FSCIL	23.53	21.73	19.83	18.62	18.45	17.34	16.36	15.27	14.48	14.34	13.86
FACT	58.45	53.14	48.77	45.05	43.97	40.57	38.77	36.62	35.60	33.86	32.71
TEEN	53.98	49.86	45.69	42.19	40.32	37.84	35.65	33.92	32.10	31.02	30.08
CLOSER	**67.35**	55.90	51.37	47.37	44.22	41.22	38.86	36.94	35.04	33.46	31.74
PFR	59.32	55.19	50.03	46.26	45.24	41.59	39.64	37.72	36.75	35.01	33.99
ADBS	56.21	50.47	46.14	42.50	39.39	36.71	34.36	32.30	30.47	29.36	27.85
Ours	60.40	**56.43**	**52.00**	**47.88**	**46.69**	**43.34**	**41.65**	**39.96**	**38.07**	**37.37**	**36.00**

Table 4: Classification accuracy on the Stanford Dog dataset.

Method	Accuracy in each session (%)								
	0	1	2	3	4	5	6	7	8
CEC	55.01	46.05	43.83	41.57	39.28	37.26	35.53	33.87	32.65
C-FSCIL	32.22	28.34	25.68	23.57	21.81	20.00	19.00	17.74	16.68
FACT	58.43	50.92	47.48	44.39	41.48	39.04	36.78	34.87	33.26
TEEN	61.09	54.70	51.71	49.56	47.28	44.41	42.30	40.56	39.14
CLOSER	60.22	53.22	49.40	46.64	43.24	40.65	38.33	36.10	34.37
PFR	59.78	51.65	48.14	45.91	41.13	40.13	38.20	37.28	36.01
ADBS	61.63	53.15	47.55	43.02	39.28	35.78	33.16	31.16	29.15
Ours	**65.57**	**58.44**	**56.03**	**53.84**	**51.34**	**49.45**	**47.42**	**45.29**	**44.06**

Table 5: Classification accuracy on the Stanford Cars dataset.

Method	Accuracy in each session (%)									
	0	1	2	3	4	5	6	7	8	9
CEC	65.23	59.22	54.43	50.75	47.57	45.38	42.41	41.16	39.17	37.09
C-FSCIL	42.80	37.50	32.83	29.63	26.76	24.64	22.82	21.57	20.18	18.90
FACT	76.76	68.57	63.29	57.93	54.59	52.69	49.77	48.02	46.12	43.98
TEEN	75.42	69.39	64.30	60.24	56.44	54.32	50.96	49.78	47.56	45.32
CLOSER	**88.56**	73.75	64.85	57.85	51.58	47.01	42.98	39.78	36.92	34.49
PFR	82.02	72.22	66.19	60.73	56.19	53.04	50.00	49.22	47.02	44.56
ADBS	75.91	64.64	56.16	49.65	44.49	40.79	37.24	34.26	32.02	29.79
Ours	85.63	**79.62**	**74.46**	**70.24**	**67.17**	**65.65**	**63.15**	**61.41**	**59.36**	**57.58**

Table 6: Classification accuracy on the FGVCAircraft dataset.

Method	Accuracy in each session (%)								
	0	1	2	3	4	5	6	7	8
CEC	50.61	43.50	40.32	37.26	34.93	33.95	32.15	30.17	29.57
C-FSCIL	56.45	48.84	43.40	39.71	36.52	33.62	31.33	29.28	28.28
FACT	55.95	49.19	45.25	41.53	38.06	35.58	34.31	32.64	31.60
TEEN	59.17	55.08	50.06	46.05	42.10	38.73	35.60	33.01	30.98
CLOSER	68.07	57.45	52.86	49.28	46.05	43.20	40.73	38.51	36.56
PFR	59.94	52.80	47.66	43.32	39.61	38.16	36.36	35.22	34.42
ADBS	59.03	53.67	50.60	46.61	44.28	41.19	39.36	36.90	35.42
Ours	**71.43**	**66.30**	**61.66**	**56.34**	**52.85**	**49.09**	**46.18**	**42.96**	**40.72**