ROBUST FEW-SHOT CLASS-INCREMENTAL LEARNING VIA ADVERSARIAL TWO-PLAYER GAME OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Few-Shot Class-Incremental Learning (FSCIL) demands that a model continuously acquire new categories from scarce labeled data while retaining prior knowledge. Recent LoRA-based methods dynamically construct lightweight adaptation modules upon Pre-Trained Models (PTMs), achieving impressive performance with minimal additional parameters. However, most existing approaches largely neglect adversarial robustness, leaving the adapted models vulnerable to adversarial threats in real-world deployment scenarios. In this work, we simultaneously address catastrophic forgetting, data scarcity, and adversarial fragility in FSCIL by introducing a new LoRA-based optimization framework that dynamically creates a dedicated LoRA expert per session, controlled by learnable layer-wise gates that adaptively balance base and incremental representations. To robustify the model under extreme data scarcity, we introduce a two-player adversarial game between a co-evolving attacker expert and a defender expert. The attacker continuously sharpens its adversarial generation capability via transferable perturbations, while the defender improves robustness by resisting both transferred and self-generated adversarial examples. We further derive that this minimax formulation admits a Nash Equilibrium and converges at a linear rate under mild smoothness conditions. To prevent representational collapse under few-shot constraints, we impose a margin-based Centered Kernel Alignment (CKA) regularization that explicitly penalizes redundancy between the base and incremental experts, steering each new expert toward a complementary feature subspace and sustaining meaningful adversarial co-evolution.

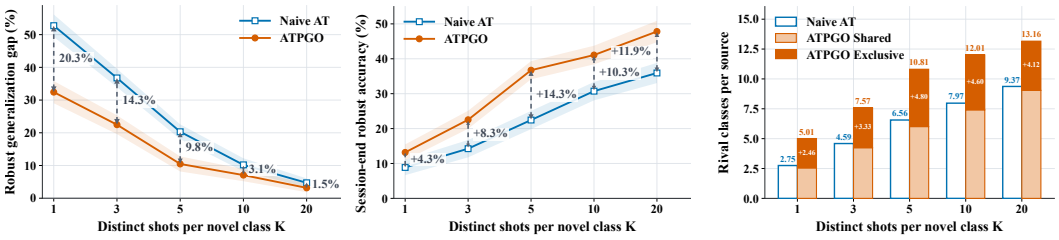
1 INTRODUCTION

Deep neural networks have achieved remarkable success across a broad spectrum of visual recognition tasks Krizhevsky et al. (2012). However, deploying such models in real-world scenarios demands two critical yet often conflicting capabilities: *continual adaptability* representing the ability to incrementally acquire knowledge of new categories without forgetting prior ones Parisi et al. (2019) and *adversarial robustness* representing the ability to maintain reliable predictions under maliciously crafted perturbations Yin et al. (2025). While each challenge has been studied extensively in isolation, their intersection remains largely unexplored, leaving a critical gap in the deployment of reliable lifelong learning systems Ye et al. (2026).

Few-Shot Class-Incremental Learning (FSCIL) represents one of the most demanding continual learning settings Tang et al. (2022) in which after an initial base session with abundant supervision, a model must sequentially incorporate novel categories from only a handful of labeled examples per class, all without access to any historical data. This data scarcity simultaneously exacerbates two well-known failure modes: (1) *catastrophic forgetting* of old classes Kang et al. (2022) and (2) *overfitting* to the new ones Srivastava et al. (2014), making FSCIL already highly challenging under standard evaluation. However, existing FSCIL methods are primarily evaluated on clean inputs and largely overlook adversarial perturbations. Even a state-of-the-art FSCIL model trained without adversarial considerations collapses in performance under standard attacks, revealing a fundamental brittleness that precludes safe deployment Xie et al. (2017); Mi et al. (2025).

A natural remedy is to apply naive adversarial training (AT) Bagnall et al. (2017) directly within the FSCIL pipeline. Yet this approach faces a fundamental dilemma: effective AT typically relies

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Robust Generalization Gap (b) Session-End Robust Accuracy (c) Rival-Class Coverage

Figure 1: **Analysis of few-shot adversarial overfitting and rival exposure.** (a) and (b) report the average difference between robust accuracy on training and test samples and the post-session robustness for CUB-200. At small K, Naive AT exhibits a large gap and low robustness, revealing adversarial overfitting. (c) shows that the single trajectory of Naive AT exposes fewer rivals, whereas ATPGO broadens rival coverage through complementary adversarial supervision. Detailed definitions and analyses are provided in the **Appendix D**.

on large and diverse training samples to generate sufficiently varied adversarial examples Mi et al. (2025), whereas FSCIL incremental sessions provide only a few labeled samples per class Tang et al. (2022). Fig. 1 (a) and (b) show that Naive AT exhibits a large robust generalization gap and low post-session robust accuracy on test samples at small K-shot. This indicates that adversarial fine-tuning achieves substantially higher robustness on the few observed training examples than on unseen samples. The robust generalization gap widens as K decreases. We term this phenomenon adversarial overfitting under data scarcity. Although recent works attempt to improve the robustness of continual learning models Mi et al. (2025); Zhou & Hua (2024) through data replay, they focus on standard continual learning settings with plentiful data and are not directly applicable to FSCIL.

This observation motivates a fundamental question: *When training data are scarce, how can a lifelong learner acquire sufficient adversarial diversity to achieve robust adaptation?* As reported in Fig. 1 (c), we empirically observe that the performance degeneration of conventional adversarial training is due to a severe adversarial diversity bottleneck in few-shot incremental learning. The generated adversarial examples concentrate on only a limited subset of rival-class directions. This limitation stems from the static attack generation process, where the model repeatedly explores its own local vulnerabilities but lacks sufficient supervision to discover diverse adversarial modes. These observations suggest that robust FSCIL requires an adaptive mechanism that can dynamically expand adversarial exposure beyond the limited trajectories induced by scarce data. From these empirical results, we argue that robustness under extreme data scarcity should not rely solely on additional samples, but rather on adaptive diversity generation during learning. Instead of treating adversarial examples as static perturbations generated from a fixed model, we view robust FSCIL as a dynamic interaction between an evolving attacker and an adaptive defender. Through continuous competition, the attacker explores previously unknown weaknesses of the defender, while the defender progressively strengthens its decision boundaries. In this way, optimization dynamics themselves become a source of diversity, compensating for the lack of training samples.

Based on this principle and empirical results, we propose FSCIL-AT, a robust few-shot class-incremental learning framework that introduces adversarial co-evolution into parameter-efficient continual adaptation. The core component of FSCIL-AT is an Adversarial Two-Player Game Optimization (ATPGO) mechanism, which formulates robustness learning as a minimax game between a defender expert and a co-evolving attacker expert. The attacker continuously adapts its adversarial generation strategy by probing the defender's current vulnerabilities, producing transferable adversarial examples that provide increasingly challenging supervision. Meanwhile, the defender learns to resist both transferred and self-generated attacks, enabling robust adaptation without requiring additional data. We further provide theoretical analysis showing that under mild smoothness assumptions, the proposed game formulation admits a Nash equilibrium and alternating optimization converges toward this equilibrium.

To prevent the incremental expert from collapsing into a near-copy of the base expert under data scarcity, which would simultaneously waste representational capacity and degenerate the adversarial game into a trivial one, we introduce a Margin-based Centered Kernel Alignment (CKA) Regularization (MBCKAR) that explicitly penalizes representational redundancy between experts and steers each incremental expert toward a genuinely complementary feature subspace. We construct a

series of experiments and the empirical results demonstrate that our FSCIL-AT achieves the state-of-the-art performance. The contributions of this paper are summarized as: (1) We identify and formally characterize a critical yet overlooked vulnerability: state-of-the-art FSCIL models catastrophically collapse under adversarial perturbations. To the best of our knowledge, this is the first work to systematically address adversarial robustness in the few-shot class-incremental learning setting; (2) We propose an adversarial two-player game optimization framework, where an adaptive attacker and defender continuously interact to generate and overcome evolving adversarial challenges, effectively replacing unavailable data diversity with optimization-driven diversity; (3) We propose a margin-based hinge CKA regularization that explicitly penalizes representational redundancy between the base and incremental experts, steering each incremental expert toward a genuinely complementary feature subspace and sustaining meaningful attacker–defender co-evolution throughout training; (4) We prove equilibrium existence and convergence properties of the proposed game formulation and demonstrate state-of-the-art performance.

2 Related Work

Few-Shot Class-Incremental Learning. FSCIL learns novel classes from scarce labeled examples while retaining prior knowledge, a setting formalized by Tao et al. (2020). Subsequent methods mainly developed along two directions. The first conducts classifier adaptation on frozen representations Zhang et al. (2021); Hersche et al. (2022); Zhuang et al. (2023); Wang et al. (2023); Chen et al. (2025). The second direction focuses on enhancing feature separability across sessions Ahmed et al. (2024); Song et al. (2023); Zhao et al. (2023); Park et al. (2024); Liu & Yang (2025); Shi et al. (2025); Zhou et al. (2022); Wang et al. (2026a); Xie et al. (2025). Nevertheless, these methods assume benign inputs and do not address adversarial vulnerability during continual adaptation.

Adversarially robust continual learning. Adversarial training improves robustness against worst-case perturbations, with early studies focusing on stationary settings Zhang et al. (2019); Wang et al. (2019); Gowal et al. (2019); Chen et al. (2020); Zheng et al. (2020); Hua et al. (2024); Chopra et al. (2026). Recent work extends adversarial robustness to continual learning Zhou & Hua (2024); Cho et al. (2025); Ye et al. (2026); Wang et al. (2026b); Bonicelli et al. (2022); Mi et al. (2025); Dai et al. (2025). Further discussion is provided in **Appendix A**.

3 Methodology

3.1 Problem Formulation

We study the model's robustness under FSCIL. Formally, let $\mathcal{S} = \{S_0, S_1, \ldots, S_T\}$ denote a sequence of $T+1$ learning sessions. The base session S_0 provides abundant labeled data $\mathcal{D}_0 = \{(\mathbf{x}_i, y_i)\}$ covering C_0 base classes. Each subsequent incremental session S_t ($t > 0$) introduces C_t novel classes disjoint from all prior classes, with only K labeled support samples per class. The class sets across all sessions are mutually exclusive: $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$ for $i \neq j$. After learning session S_t, the model is evaluated on all classes seen so far $\mathcal{C}_{0:t} = \bigcup_{i=0}^{t} \mathcal{C}_i$, without access to any historical training data. In addition, we also evaluate the robust performance of various continual learning models using adversarial attack methods.

3.2 Model Architecture

Following recent works He et al. (2025a), our architecture builds upon a pre-trained Vision Transformer Dosovitskiy et al. (2020). As shown in Fig. 2, FSCIL-AT integrates session-wise LoRA expert adaptation, adversarial two-player game optimization, and margin-based CKA regularization into a unified continual-learning pipeline. The backbone parameters θ_{ViT} are frozen throughout all sessions, preserving the general visual representations acquired during pre-training and preventing catastrophic forgetting of the feature extractor. For each session t, we create a dedicated LoRA expert $\mathcal{E}_t$ consisting of low-rank adaptation matrices injected into every Transformer block of the backbone. Specifically, for each Transformer block $l \in \{1, \ldots, L\}$, the LoRA residual is computed as $\Delta_t^l(\mathbf{h}) = \mathbf{B}_t^l \mathbf{A}_t^l \mathbf{h}$, where $\mathbf{h} \in \mathbb{R}^D$ is the input token representation, $\mathbf{A}_t^l \in \mathbb{R}^{r \times D}$ and $\mathbf{B}_t^l \in \mathbb{R}^{D \times r}$ are the low-rank factors with rank $r \ll D$, where D is the input dimension.

Layer-wise Gating for Incremental Sessions. In the base session ($t = 0$), we create the base expert $\mathcal{E}_0$ consisting of L LoRA modules. The l-th layer output of the expert $\mathcal{E}_0$ is defined as $\mathbf{z}_0^l = f_{\mathrm{ViT}}^l(\mathbf{h}_0^l) + \Delta_0^l(\mathbf{h}_0^l)$, where $f_{\mathrm{ViT}}^l(\cdot)$ denotes the frozen ViT mapping at layer l. In incremental sessions ($t > 0$), the model leverages both universal representations from the base expert $\mathcal{E}_0$ and task-specific representations from the current expert $\mathcal{E}_t$. To adaptively balance their contributions, we introduce a learnable layer-wise gate $g_t^l \in [0, 1]$ for each Transformer block l, yielding $\mathbf{z}_t^l = f_{\mathrm{ViT}}^l(\mathbf{h}_t^l) + g_t^l \cdot \Delta_0^l(\mathbf{h}_t^l) + \Delta_t^l(\mathbf{h}_t^l)$, where $g_t^l = \sigma(\alpha_t^l)$, α_t^l is a trainable scalar gate logit, and $\sigma(\cdot)$

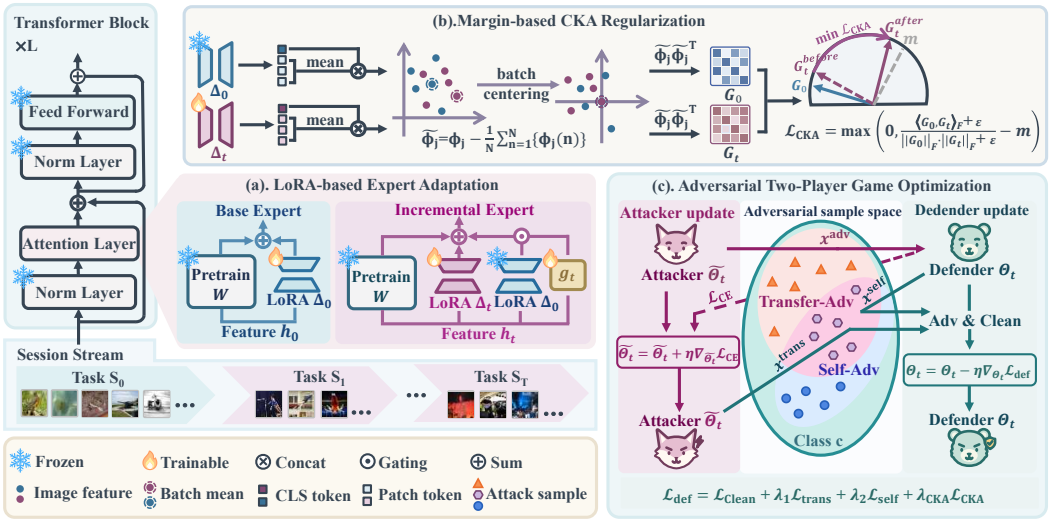


Figure 2: **Overview of the proposed FSCIL-AT framework. (A) LoRA-based expert adaptation.** The base session learns a reusable LoRA expert, adaptively gated with a trainable incremental expert in each session. **(B) Adversarial two-player game optimization.** The attacker is updated against the fixed defender to expose vulnerabilities and generate transferred attacks. The defender learns from clean, transferred, and self-generated adversarial examples, yielding stronger, more diverse robust supervision. **(C) Margin-based CKA regularization.** CKA constrains excessive similarity between the two experts, promoting complementarity while preserving shared knowledge.

is the sigmoid function. All LoRA injection positions within the same Transformer block share a single gate coefficient. The base expert parameters $\{\mathbf{A}_0^l, \mathbf{B}_0^l\}$ are frozen in all incremental sessions and only $\{\mathbf{A}_t^l, \mathbf{B}_t^l\}$ and $\{\alpha_t^l\}$ are updated to adapt to new tasks.

Classification Heads. Each session expert $\mathcal{E}_t$ is paired with a local cosine classifier $W_t = [(w_t^c)^\top]_{c\in\mathcal{C}_t} \in \mathbb{R}^{C_t\times D}$, where w_t^c denotes the classifier weight vector for class c. The cosine logit for class c is defined by $s_t^c(\mathbf{x}) = \tau \frac{f_t(\mathbf{x})^\top w_t^c}{\|f_t(\mathbf{x})\|_2 \|w_t^c\|_2}$, where $f_t(\mathbf{x})$ is the feature extracted by expert $\mathcal{E}_t$ and τ is a learnable temperature. The parametric classifier is discarded after session training and only serves as an optimization surrogate. Following He et al. (2025a), we maintain a non-parametric global prototype classifier to preserve all previous category information used for inference. For each class c belonging to session t, the prototype is computed from the set $\mathcal{X}_c^t$ of original training images as $\boldsymbol{\mu}_c^t = \frac{1}{|\mathcal{X}_c^t|}\sum_{\mathbf{x}\in\mathcal{X}_c^t} \frac{f_t(\mathbf{x})}{\|f_t(\mathbf{x})\|_2}$. Prototypes are computed once after training and never modified when new sessions arrive. The growing prototype set after session t is $\mathcal{M}_t = \mathcal{M}_{t-1} \cup \{\mu_c^t\}_{c\in\mathcal{C}_t}$. At test time, for a query image $\mathbf{x}$, each learned expert $\mathcal{E}_i$ $(i = 0, \ldots, t)$ independently extracts a feature $f_i(\mathbf{x})$. The final prediction for a given $\mathbf{x}$ is defined as:

$$y' = \underset{i=\{0,\cdots,t\}, c\in\mathcal{C}_i}{\operatorname{argmax}} \cos(\boldsymbol{\mu}_c^i, f_i(\mathbf{x})), \tag{1}$$

where $\cos(\cdot,\cdot)$ denotes the cosine similarity and y' is the predicted class label. Eq. (1) allows the proposed framework to make prediction without task information at the inference phase.

3.3 Adversarial Two-Player Game Training

Standard adversarial training typically requires large datasets to be effective. In the FSCIL setting, where incremental sessions contain only K samples per class, naive adversarial fine-tuning risks severe overfitting. To address this issue, we propose a two-player adversarial game optimization process between a defender expert (the current session expert $\mathcal{E}_t$) and a co-evolving attacker expert $\tilde{\mathcal{E}}_t$, which actively discovers the defender's current vulnerabilities and provides transferable adversarial supervision throughout training, as shown in Fig. 2 (c).

Attacker Expert Initialization. At session t, we first optimize the current expert $\mathcal{E}_t$ on the training dataset to acquire category recognition capability, which we refer to as the clean warm-up phase. After this optimization phase, we construct an attacker expert $\tilde{\mathcal{E}}_t$ by cloning the LoRA parameters of the current expert, while the defender is instantiated as the current active expert $\mathcal{E}_t$. The attacker

shares the same frozen backbone θ_{ViT}, frozen base expert $\mathcal{E}_0$, current gate logits $\{\alpha_t^l\}$, and frozen global prototype classifier with the defender. The only difference between them lies in the independently updated LoRA residuals of the attacker during the subsequent optimization process.

Two-Player Game Optimization. We formulate the attacker and defender as two players in a game-based optimization framework, where the attacker's objective is to generate high-quality adversarial examples that hinder the defender from correctly recognizing their labels, while the defender aims to identify and resist the adversarial samples generated by the attacker. These two models compete with each other throughout the optimization process, driving the defender to continuously enhance its adversarial defense capability. Specifically, we formulate this adversarial two-player game as a minimax optimization problem over the joint parameter space $(\tilde{\Theta}_t, \Theta_t) \in \mathcal{A} \times \mathcal{D}$. The unified optimization objective is defined as:

$$\min_{\Theta_t \in \mathcal{D}} \max_{\tilde{\Theta}_t \in \mathcal{A}} \mathcal{J}(\tilde{\Theta}_t, \Theta_t) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}\Big[\underbrace{\mathcal{L}_{\text{CE}}\Big(\mathcal{E}_t, F_g(\mathbf{x}, \tilde{\mathcal{E}}_t), y\Big)}_{\text{(I) Transferability Term}} + \underbrace{\mathcal{L}_{\text{CE}}\Big(\mathcal{E}_t, F_g(\mathbf{x}, \mathcal{E}_t), y\Big)}_{\text{(II) Self-Vulnerability Term}}\Big], \quad (2)$$

where $\tilde{\Theta}_t$ and Θ_t denote the LoRA parameters of $\tilde{\mathcal{E}}_t$ and $\mathcal{E}_t$. $\mathcal{A}$ and $\mathcal{D}$ are the strategy spaces of the attacker and defender. $F_g(\mathbf{x}, \tilde{\mathcal{E}}_t)$ is the adversarial sample generation function, expressed as:

$$F_g(\mathbf{x}, \tilde{\mathcal{E}}_t) = \Pi_{\mathcal{B}_\infty(\mathbf{x},\epsilon)\cap[0,1]^{3HW}}\left(\mathbf{x} + \alpha\, r^\star\right), \quad r^\star \in \underset{\|r\|_\infty \le 1}{\arg\max} \left\langle r,\, \nabla_{\mathbf{x}} \mathcal{L}_{\text{CE}}\big(\tilde{\mathcal{E}}_t, \mathbf{x},\, y\big)\right\rangle, \quad (3)$$

where $\mathbf{x}^{\text{adv}} = F_g(\mathbf{x}, \tilde{\mathcal{E}}_t)$ denotes the adversarial example crafted from the clean input $\mathbf{x}$, H and W are the image height and width, respectively, and $3HW$ corresponds to the total number of pixel values across all three RGB channels. $\Pi_{\mathcal{B}_\infty(\mathbf{x},\epsilon)\cap[0,1]^{3HW}}(\cdot)$ is the projection operator onto the feasible set, which is the intersection of the ℓ_∞ ball $\mathcal{B}_\infty(\mathbf{x}, \epsilon) = \{\mathbf{x}' \mid \|\mathbf{x}' - \mathbf{x}\|_\infty \le \epsilon\}$ of radius $\epsilon > 0$ centered at $\mathbf{x}$ and the valid pixel range $[0, 1]^{3HW}$, ensuring that the perturbed image remains both imperceptible and photometrically valid. $\alpha > 0$ is the perturbation step size controlling the magnitude of the gradient update. $r^\star$ is the optimal unit-bounded perturbation direction and $\|\cdot\|_\infty$ denotes the ℓ_∞ norm and the constraint $\|r\|_\infty \le 1$ restricts the search to the unit ℓ_∞ ball so that α solely governs the overall step magnitude. $\mathcal{L}_{\text{CE}}\big(\tilde{\mathcal{E}}_t, \mathbf{x},\, y\big)$ is the cross-entropy loss function calculated using the attack expert $\tilde{\mathcal{E}}_t$.

$$\mathcal{L}_{\text{CE}}\big(\tilde{\mathcal{E}}_t, \mathbf{x},\, y\big) = -\sum\nolimits_{c=1}^{\mathcal{C}_t} \left\{F_{\text{eq}}(y, c) \log \tilde{s}_t^c(\mathbf{x})\right\}, \quad (4)$$

where $\tilde{s}_t^c(\mathbf{x})$ denotes the c-th logit produced by the attack expert $\tilde{\mathcal{E}}_t$ and $\mathcal{C}_t$ is the number of categories at the session t. $F_{\text{eq}}(c, y)$ denotes a function that returns one if $c = y$, otherwise, return zero.

The maximization over $\tilde{\Theta}_t$ drives the attacker to craft maximally harmful adversarial examples for the defender, and the minimization over Θ_t drives the defender to resist such attacks. Term (I) in Eq. (2) is the *transferability term*: it couples the two players by measuring the cross-entropy loss of the defender $\mathcal{E}_t$ on adversarial examples $\mathbf{x}^{\text{trans}}$ generated by the attacker $\tilde{\mathcal{E}}_t$. The attacker maximizes this term by sharpening its adversarial generation capability $\tilde{\Theta}_t \leftarrow \tilde{\Theta}_t + \eta\, \nabla_{\tilde{\Theta}_t} \mathcal{J}$, while the defender minimizes it by improving robustness $\Theta_t \leftarrow \Theta_t - \eta\, \nabla_{\Theta_t} \mathcal{J}$. Term (II) in Eq. (2) is the *self-vulnerability term*: since $\mathbf{x}^{\text{trans}}$ only partially captures the defender's local decision boundary, we additionally generate $\mathbf{x}^{\text{self}} = F_g(\mathbf{x}, \mathcal{E}_t)$. Optimized only with respect to Θ_t, this term leaves the attacker's payoff unchanged, preserving the equilibrium of the transfer game while providing complementary robustness regularizer. Based on Eq. (2), the optimization is decomposed into the following two stages.

Stage 1: Attacker Update. At this stage, we freeze all parameters of the defender and employ the attacker expert to generate an adversarial example $\mathbf{x}^{\text{adv}} = F_g(\mathbf{x}, \tilde{\mathcal{E}}_t)$ of a given sample $\mathbf{x}$. The temporary adversarial example $\mathbf{x}^{\text{adv}}$ is then fed into the current defender expert $\mathcal{E}_t$, and the attacker loss is defined to quantify the degree of difficulty imposed on the defender, measured by the negative cross-entropy of the defender's prediction on $\mathbf{x}^{\text{adv}}$, expressed as $\mathcal{L}_{\text{CE}}\big(\mathcal{E}_t, \mathbf{x}^{\text{adv}},\, y\big)$. The attacker aims to increase the classification loss of the defender. Therefore, its parameter updating is defined as:

$$\tilde{\Theta}_t = \tilde{\Theta}_t + \eta \nabla_{\tilde{\Theta}_t} \mathcal{L}_{\text{CE}}\big(\mathcal{E}_t, \mathbf{x}^{\text{adv}},\, y\big), \quad (5)$$

where η is the learning rate and $\nabla_{\tilde{\Theta}_t}\mathcal{L}_{\text{CE}}\big(\mathcal{E}_t, \mathbf{x}^{\text{adv}}, y\big)$ is the gradient vector.

Stage 2: Defender Update. Since the temporary adversarial example $\mathbf{x}^{\text{adv}}$ generated in the preceding stage was produced by the pre-update attacker expert, it primarily serves to construct a differentiable optimization path for the attacker and therefore does not fully reflect the attacker's latest attack capability. Consequently, we regenerate a transfer adversarial example $\mathbf{x}^{\text{trans}} = F_g(\mathbf{x}, \tilde{\mathcal{E}}_t)$ using the post-update attacker expert, so as to faithfully capture the attacker's evolved adversarial strength after completing the current round of updates. In addition, we also employ the defender expert to produce the adversarial generated sample $\mathbf{x}^{\text{self}} = F_g(\mathbf{x}, \mathcal{E}_t)$, which characterizes the local vulnerability directions in the vicinity of its current decision boundary. Based on these adversarial samples, the defender's joint loss is defined as:

$$\mathcal{L}_{\text{def}} = \mathcal{L}_{\text{CE}}(\mathcal{E}_t, \mathbf{x}, y) + \lambda_1\mathcal{L}_{\text{CE}}(\mathcal{E}_t, \mathbf{x}^{\text{trans}}, y) + \lambda_2\mathcal{L}_{\text{CE}}(\mathcal{E}_t, \mathbf{x}^{\text{self}}, y), \tag{6}$$

where λ_1 and λ_2 are two hyperparameters that control the importance of $\mathbf{x}^{\text{trans}}$ and $\mathbf{x}^{\text{self}}$ during optimization.

Despite adversarial robustness gained from two-player training, scarce labeled data in incremental sessions (K samples per class) may drive $\mathcal{E}_t$ to mimic representations of the frozen expert $\mathcal{E}_0$, reducing expert diversity. We therefore introduce a CKA regularization term $\mathcal{L}_{\text{CKA}}$ (Section 3.4) to promote complementary feature learning between $\mathcal{E}_0$ and $\mathcal{E}_t$.

3.4 Complementarity Constraint via CKA Regularization

While adversarial training improves robustness, it does not guarantee complementary representations between $\mathcal{E}_t$ and $\mathcal{E}_0$. With limited incremental data (K samples per class), the gating mechanism may cause $\mathcal{E}_t$ to mimic $\mathcal{E}_0$, wasting incremental capacity and weakening adversarial diversity. Since L_2 feature distance is sensitive to scale and rotation, we adopt Centered Kernel Alignment (CKA) Kornblith et al. (2019) to measure invariant representation similarity and encourage complementary feature learning at LoRA injection points.

Position-wise Residual Features. At layer l and injection position $p \in \mathcal{P}$, where $\mathcal{P}$ is the index set of linear projection sublayers within a Transformer block, for a mini-batch of N images, the base and task expert LoRA residuals are calculated by:

$$\mathbf{R}_0^{l,p} = \mathbf{B}_0^{l,p}\mathbf{A}_0^{l,p}\mathbf{H}^{l,p}, \quad \mathbf{R}_t^{l,p} = \mathbf{B}_t^{l,p}\mathbf{A}_t^{l,p}\mathbf{H}^{l,p}, \tag{7}$$

where $\mathbf{H}^{l,p} \in \mathbb{R}^{D\times N(1+P)}$ stacks the input token representations across the batch (P is the number of patch tokens). To obtain a compact per-image representation that captures both global semantics and local visual responses without constructing a prohibitively large Gram matrix, we extract and concatenate the CLS token residual and the mean patch token residual for each image $\mathbf{x}$:

$$\mathbf{\Phi}_j^{\mathbf{x}} = \big[\mathbf{R}_j^{l,p}[\texttt{CLS}]^x \otimes \overline{\mathbf{R}}_j^{l,p}[\texttt{patch}]^x\big] \in \mathbb{R}^{2D}, \quad j \in \{0, t\}, \tag{8}$$

where $\overline{\mathbf{R}}_j^{l,p}[\texttt{patch}]^x$ denotes the mean over all patch token residuals of image $\mathbf{x}$ and $\otimes$ denotes the concatenation along the feature dimension. The residual of the CLS token primarily captures information related to image-level semantics and classification decisions, while the averaged patch residual summarizes the collective response of the experts across all local image regions. Concatenating the two preserves both global semantics and local visual information, while avoiding the construction of a massive Gram matrix by treating all $B \times N$ tokens as independent samples. Subsequently, the concatenated representations from the two experts are separately centered along the batch dimension:

$$\tilde{\mathbf{\Phi}}_j = \mathbf{\Phi}_j - \frac{1}{N}\sum\nolimits_{n=1}^{N}\big\{\mathbf{\Phi}_j(n)\big\}, \quad j \in \{0, t\}, \tag{9}$$

where $\mathbf{\Phi}_j(n)$ denotes the n-th element of $\mathbf{\Phi}_j$. Based on Eq. (9), the sample Gram matrices are defined as $G_j = \tilde{\mathbf{\Phi}}_j^\top\tilde{\mathbf{\Phi}}_j \in \mathbb{R}^{N\times N}$. The Linear CKA similarity between the two experts at position (l, p) is then defined as:

$$\text{CKA}(G_0, G_t) = \frac{\langle G_0, G_t\rangle_F + \varepsilon}{\|G_0\|_F \cdot \|G_t\|_F + \varepsilon}, \tag{10}$$

where $\langle\cdot,\cdot\rangle_F = \text{tr}(G_0^\top G_t)$ is the Frobenius inner product and ε is a numerical stability constant.

Table 1: **Clean and adversarial accuracy on CUB-200.** Top-1 accuracy (%) is reported as mean±std. **Bold** and underlined values indicate the best and second-best results, respectively. Avg-All is the unweighted mean of clean accuracy and the five adversarial accuracies.

Method	Train	Clean	Robust accuracy					Avg-All
			FGSM	ℓ_∞-PGD	ℓ_2-PGD	BIM	CW	
LwF (ECCV'16)	Std	62.93±00.30	08.96±00.21	01.29±00.13	13.32±00.32	01.10±00.08	00.00±00.00	14.60
	Adv	62.37±00.20	13.60±00.19	00.98±00.00	18.20±00.19	00.95±00.10	00.00±00.00	16.02
RanPAC (NeurIPS'23)	Std	43.56±08.53	03.15±02.70	00.00±00.00	11.46±09.93	00.00±00.00	01.13±00.98	09.88
	Adv	34.60±09.89	09.74±00.17	00.00±00.00	27.29±00.35	00.00±00.00	13.70±01.09	14.22
CL-LoRA (CVPR'25)	Std	80.95±00.10	10.08±00.36	00.00±00.00	34.08±00.19	00.00±00.00	25.84±00.53	25.16
	Adv	80.53±00.31	27.36±01.93	00.04±00.04	51.68±01.49	00.05±00.06	49.13±01.61	34.80
KeepLoRA (ICLR'26)	Std	71.84±00.53	15.12±00.50	00.36±00.21	25.61±00.59	00.25±00.11	11.01±00.46	20.70
	Adv	71.08±00.45	29.92±00.78	**02.60±00.12**	40.71±01.02	02.36±00.19	25.52±00.88	28.70
TEEN (NeurIPS'23)	Std	76.50±00.00	08.33±00.03	01.73±00.06	11.52±00.09	01.58±00.03	00.00±00.00	16.61
	Adv	76.50±00.00	08.33±00.03	01.78±00.12	11.52±00.09	01.58±00.03	00.00±00.00	16.62
BiMC (CVPR'25)	Std	66.02±00.00	03.14±00.00	00.00±00.00	04.62±00.00	00.00±00.00	04.47±00.07	13.04
	Adv	65.00±00.08	04.21±00.11	00.00±00.00	06.31±00.05	00.00±00.00	05.30±00.26	13.47
ConCM (ICLR'26)	Std	80.02±00.32	07.02±00.15	00.00±00.00	17.29±00.43	00.00±00.00	02.26±00.12	17.77
	Adv	79.55±00.09	06.59±00.14	00.00±00.00	18.11±00.48	00.00±00.00	02.10±00.08	17.73
AIR (CVPR'24)	Native	40.83±00.92	07.00±00.43	01.51±00.33	08.85±00.64	01.37±00.30	00.00±00.00	09.93
DSEF (NeurIPS'25)	Native	18.50±00.20	06.42±00.17	00.65±00.24	00.64±00.17	00.64±00.17	00.58±00.21	04.57
ARMCL (ICCV'25)	Native	46.22±02.27	18.65±00.71	01.73±00.57	19.65±00.71	01.53±00.82	00.00±00.00	14.63
FSCIL-AT (Ours)	**ATPGO**	**82.20±00.30**	**33.89±00.23**	02.36±01.50	**51.81±01.23**	**02.38±01.47**	**50.52±00.24**	**37.19**

Margin-based CKA Loss. Since the incremental expert still benefits from sharing general visual knowledge with the base expert, we do not drive CKA to zero. Instead, we adopt a margin-based hinge formulation that only penalizes redundancy exceeding a tolerance threshold m:

$$\mathcal{L}_{\text{CKA}}^{l,p} = \max\Big(0,\ \text{CKA}(G_0^{l,p}, G_t^{l,p}) - m\Big). \tag{11}$$

The total CKA loss for session t averages over the selected layer set $\mathcal{L}_{\text{CKA}}$ and all injection positions:

$$\mathcal{L}_{\text{CKA}} = \frac{1}{L \cdot |\mathcal{P}|} \sum\nolimits_{l \in \mathcal{L}_{\text{CKA}}} \sum\nolimits_{p \in \mathcal{P}} \mathcal{L}_{\text{CKA}}^{l,p}, \tag{12}$$

where $|\mathcal{P}|$ denote the number of positions. This averaging prevents the loss magnitude from scaling with the number of monitored layers or positions, enabling stable joint optimization. By integrating the $\mathcal{L}_{\text{CKA}}$ loss, the main objective function defined in Eq. (6) is expressed as:

$$\mathcal{L}_{\text{def}} = \mathcal{L}_{\text{CE}}(\mathcal{E}_t, \mathbf{x}, y) + \lambda_1 \mathcal{L}_{\text{CE}}(\mathcal{E}_t, \mathbf{x}^{\text{trans}}, y) + \lambda_2 \mathcal{L}_{\text{CE}}(\mathcal{E}_t, \mathbf{x}^{\text{self}}, y) + \lambda_{\text{CKA}} \mathcal{L}_{\text{CKA}}, \tag{13}$$

where the CKA term is included only for incremental sessions ($t > 0$). λ_{CKA} is a hyperparameter that controls the importance of the $\mathcal{L}_{\text{CKA}}$ loss during the optimization process.

3.5 Theoretical Analysis of the Two-Player Game

Game-Theoretic Formulation. We model the adversarial interaction between the attacker expert $\tilde{\mathcal{E}}_t$ (parameterized by $\tilde{\Theta}_t$) and the defender expert $\mathcal{E}_t$ (parameterized by Θ_t) as a two-player zero-sum game $\mathcal{G} = (\mathcal{A}, \mathcal{D}, u)$, where $\mathcal{A}$ and $\mathcal{D}$ denote the strategy spaces of the attacker and defender, respectively, and u is the shared payoff functional. More theoretical results are provided in Appendix B.

Definition 1 (Strategy Spaces). The attacker's strategy space is $\mathcal{A} = \{\tilde{\Theta}_t \mid \tilde{\Theta}_t \in \mathbb{R}^d\}$, where d is the total dimension of the LoRA parameter space. The defender's strategy space is $\mathcal{D} = \{\Theta_t \mid \Theta_t \in \mathbb{R}^d\}$. For a given pair $(\tilde{\Theta}_t, \Theta_t) \in \mathcal{A} \times \mathcal{D}$ and a clean sample $(\mathbf{x}, y)$, the attacker generates an adversarial example via $\mathbf{x}^{\text{adv}} = F_g(\mathbf{x}, \tilde{\mathcal{E}}_t)$.

Definition 2 (Payoff Functional). The payoff functional $u : \mathcal{A} \times \mathcal{D} \to \mathbb{R}$ is defined as:

$$u(\tilde{\Theta}_t,\ \Theta_t)\ =\ \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}\Big[\mathcal{L}_{\text{CE}}\Big(\mathcal{E}_t,\ F_g(\mathbf{x}, \tilde{\mathcal{E}}_t),\ y\Big)\Big], \tag{14}$$

where $\mathcal{D}_t$ is the data distribution of session t. The attacker seeks to maximize u (increasing defender loss), while the defender seeks to minimize u (improving robustness), yielding the minimax objective $\min_{\Theta_t}\ \max_{\tilde{\Theta}_t}\ u(\tilde{\Theta}_t, \Theta_t)$.

Table 3: **Component ablation of FSCIL-AT.** Rob-Avg denotes the unweighted mean accuracy over the five adversarial attacks.

Components			CIFAR-100			mini-ImageNet			CUB-200			ImageNet-R		
LWG	ATPGO	MBCKAR	Clean	Rob-Avg	Avg-All	Clean	Rob-Avg	Avg-All	Clean	Rob-Avg	Avg-All	Clean	Rob-Avg	Avg-All
			82.46	11.94	23.69	94.94	25.65	37.20	77.41	9.29	20.64	75.70	13.99	24.28
✓			84.16	12.57	24.50	**96.13**	28.23	39.55	83.97	12.40	24.33	79.17	17.04	27.40
✓	✓		79.54	28.35	36.88	92.62	50.75	57.73	79.90	26.37	35.29	75.01	28.12	35.94
✓		✓	**85.07**	16.57	27.99	95.44	37.96	47.54	**84.02**	16.35	27.63	**81.88**	21.61	31.66
✓	✓	✓	81.83	**30.57**	**39.12**	93.80	**52.59**	**59.46**	82.20	**28.19**	**37.19**	75.92	**29.92**	**37.59**

Theorem 3.1 (Nash Equilibrium Existence) *Suppose the parameter spaces $\mathcal{A}$ and $\mathcal{D}$ are compact and convex subsets of $\mathbb{R}^d$, and that $u(\tilde{\Theta}_t, \Theta_t)$ is continuous in both arguments, convex in Θ_t, and concave in $\tilde{\Theta}_t$. These conditions are invoked as a local approximation of the non-convex landscape arising from the deep LoRA parameterization and the non-differentiable* $\arg\max$ *in F_g; they are standard in the adversarial-training literature Madry et al. (2017) and are used here to guarantee the existence of a well-defined equilibrium in the neighborhood of a stationary point. Then there exists a Nash Equilibrium $(\tilde{\Theta}_t^\star, \Theta_t^\star) \in \mathcal{A} \times \mathcal{D}$ satisfying:*

$$u\left(\tilde{\Theta}_t^\star,\, \Theta_t\right) \;\le\; u\left(\tilde{\Theta}_t^\star,\, \Theta_t^\star\right) \;\le\; u\left(\tilde{\Theta}_t,\, \Theta_t^\star\right), \quad \forall\, \tilde{\Theta}_t \in \mathcal{A},\; \Theta_t \in \mathcal{D}. \tag{15}$$

The proof is provided in **Appendix B**. Theorem 3.1 provides a theoretical justification for ATPGO by showing that the proposed minimax game is well-posed under the stated assumptions. Consequently, alternating optimization drives the attacker and defender toward a locally stable equilibrium, yielding optimization-driven adversarial diversity without requiring replay data.

4 Experiments

Due to the space limitation, we provide the experimental details in **Appendix C**.

4.1 Comparison to State-of-the-Art

Table 2: **Avg-All comparison across four benchmarks.** Methods without native adversarial training use Adv variants. $(\cdot)$ denotes the improvement of FSCIL-AT over the corresponding baseline.

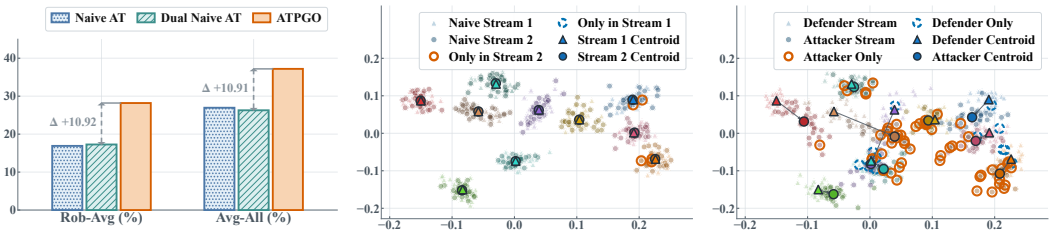
Method	CIFAR-100	mini-ImageNet	CUB-200	ImageNet-R
LwF (ECCV'16)	21.70 (+17.42)	34.86 (+24.60)	16.02 (+21.17)	13.69 (+23.90)
RanPAC (NeurIPS'23)	23.74 (+15.38)	36.00 (+23.46)	14.22 (+22.97)	24.38 (+13.21)
CL-LoRA (CVPR'25)	29.31 (+09.81)	51.31 (+08.15)	34.80 (+02.39)	35.20 (+02.39)
KeepLoRA (ICLR'26)	33.22 (+05.90)	51.44 (+08.02)	28.70 (+08.49)	32.98 (+04.61)
TEEN (NeurIPS'23)	19.33 (+19.79)	41.69 (+17.77)	16.62 (+20.57)	15.66 (+21.93)
BiMC (CVPR'25)	16.41 (+22.71)	31.73 (+27.73)	13.47 (+23.72)	21.94 (+15.65)
ConCM (ICLR'26)	18.44 (+20.68)	37.14 (+22.32)	17.73 (+19.46)	17.01 (+20.58)
AIR (CVPR'24)	13.07 (+26.05)	25.12 (+34.34)	09.93 (+27.26)	06.53 (+31.06)
DSEF (NeurIPS'25)	07.33 (+31.79)	06.45 (+53.01)	04.57 (+32.62)	02.55 (+35.04)
ARMCL (ICCV'25)	18.46 (+20.66)	37.64 (+21.82)	14.63 (+22.56)	11.52 (+26.07)
FSCIL-AT (Ours)	**39.12** (0.00)	**59.46** (0.00)	**37.19** (0.00)	**37.59** (0.00)

Tables 1 and 2 show that FSCIL-AT achieves the best aggregate performance across all benchmarks while leading under most attacks. Adversarial training lowers clean accuracy in non-adversarial baselines by imposing stronger decision-boundary constraints. TEEN's frozen representation and training-free incremental calibration leave little scope for adversarial adaptation, producing nearly identical Std and Adv results. CL-LoRA and KeepLoRA remain competitive by structuring low-rank updates to reduce interference while retaining plasticity. However, self-adversarial training exposes only model-specific vulnerabilities. FSCIL-AT instead combines adaptive base expert reuse, complementary incremental features, and co-evolved transferred and self-generated adversaries, enabling broader boundary exploration and the strongest aggregate performance. Additional comparative results are provided in **Appendix E**.

4.2 Analysis

Ablation Studies of FSCIL-AT. Table 3 evaluates the proposed components. The LWG denotes the learnable layer-wise gating mechanism. Without LWG, each incremental gate is fixed at 0.5; without ATPGO, adversarial training is disabled. Learnable gating balances reusable base knowledge with session specific representations, improving accuracy and the tradeoff between stability and plasticity. Adding ATPGO markedly improves Rob-Avg but slightly reduces clean accuracy because transferred and self generated adversarial examples impose stronger constraints on scarce clean samples, nevertheless, Avg-All increases substantially. MBCKAR further improves both clean and robust performance, and the complete model achieves the best results, raising Avg-All over the baseline without these components by 13.31% to 22.26%.

Effectiveness of ATPGO. To isolate adversarial co-evolution from sample count, we compare three settings: Naive AT uses one conventional attack stream; Dual Naive AT duplicates it to match



(a) Quantitative comparison (b) Dual Naive AT Projection (c) ATPGO Projection

Figure 3: **Comparison of adversarial example generation strategies.** Results on CUB-200, with feature projections shown at $K = 5$ in session 5.

ATPGO's adversarial-sample budget; and ATPGO combines a Defender Stream with a co-evolving Attacker Stream. Fig. 3 (a) shows that duplicating Naive AT yields little gain, whereas ATPGO outperforms Dual Naive AT on both Rob-Avg and Avg-All. Fig. 3 (b) and (c) compare adversarial feature distributions. In the shared 2D projection, colors and markers encode classes and streams, large markers indicate class centroids, and rings highlight stream-exclusive rivals. In (b), the two Naive AT streams largely overlap, with nearly coincident centroids and few stream-exclusive rivals, indicating redundant exploration. In (c), the Defender and Attacker Centroids are clearly separated, while Attacker Only reveals more unique rivals. Thus, co-evolution broadens attack directions and rival-class coverage, yielding complementary rather than merely additional adversarial supervision.

Effectiveness of MBCKAR. To reveal MBCKAR's mechanism, we analyze how its margin regulates representation similarity. Fig. 4 contrasts unconstrained No-CKA, selective MBCKAR ($m = 0.3$), and globally applied Zero-target CKA ($m = 0$). Using identical samples, we measure CKA between expert residuals at LoRA positions before and after training. The axes show initial CKA and its change, each point averages measurements with similar initial CKA. No-CKA leaves high similarity unchanged, MBCKAR preserves sharing below the margin while suppressing redundancy above it, Zero-target CKA suppresses both. This selectivity preserves transferable structure while creating complementary expert capacity for ATPGO.

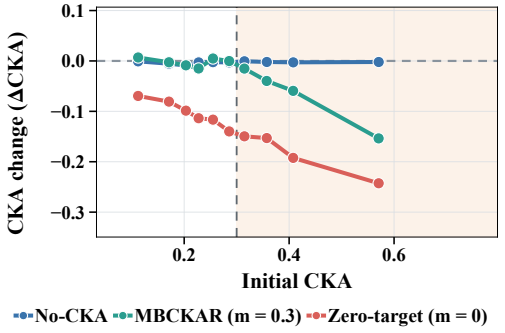


Figure 4: **Comparison of CKA-Based Expert Similarity Regulation on CUB-200.**

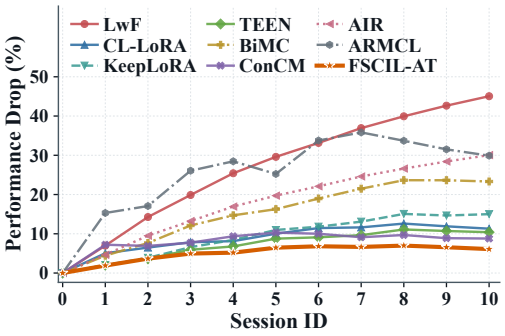


Figure 5: **Avg-All Drop from the Base Session across Baselines on CUB-200.**

Analysis of performance drop. Fig. 5 compares clean performance drop on CUB-200. For each non-adversarial continual-learning method, we report its adversarially trained variant. Among the plotted runs, FSCIL-AT maintains the smallest drop and stabilizes in later sessions, whereas LwF deteriorates progressively. TEEN declines despite frozen incremental parameters, showing that calibrated prototypes can introduce class competition without parameter drift. ARMCL's fluctuations reflect changes in the old-class replay mix induced by difficulty-based sample selection and session-specific logit calibration. As new classes compete with previously learned ones at test time, gated base reuse preserves shared cues while selective CKA curbs redundant new responses, helping limit clean performance drop.

5 Conclusion

In this paper, we propose FSCIL-AT, a robust few-shot class-incremental learning framework that addresses forgetting, data scarcity, and adversarial vulnerability. By combining LoRA-based experts, adversarial two-player optimization, and CKA regularization, FSCIL-AT achieves robust and complementary representations.

AI USE STATEMENT

Generative AI tools were used for language editing. All technical claims, implementations, experiments, and reported numbers were verified by the authors.

REFERENCES

Sravanti Addepalli, Vivek BS, Arya Baburaj, Gaurang Sriramanan, and R Venkatesh Babu. Towards achieving adversarial robustness by enforcing feature consistency across bit planes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1020–1029, 2020.

Noor Ahmed, Anna Kukleva, and Bernt Schiele. Orco: Towards better generalization via orthogonality and contrast for few-shot class-incremental learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28762–28771. IEEE, 2024.

Alexander Bagnall, Razvan Bunescu, and Gordon Stewart. Training ensembles to detect adversarial examples. *arXiv preprint arXiv:1712.04006*, 2017.

Lorenzo Bonicelli, Matteo Boschini, Angelo Porrello, Concetto Spampinato, and Simone Calderara. On the effectiveness of lipschitz-driven rehearsal in continual learning. *Advances in Neural Information Processing Systems*, 35:31886–31901, 2022.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pp. 39–57. Ieee, 2017.

Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 696–705. IEEE, 2020.

Yiyang Chen, Tianyu Ding, Lei Wang, Jing Huo, Yang Gao, and Wenbin Li. Enhancing few-shot class-incremental learning via training-free bi-level modality calibration. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9881–9890. IEEE, 2025.

Seungju Cho, Hongsin Lee, and Changick Kim. Enhancing robustness in incremental learning with adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 2518–2526, 2025.

Shivang Chopra, Shaunak Halbe, Chengyue Huang, Brisa Maneechotesuwan, and Zsolt Kira. The geometry of robustness: Optimizing loss landscape curvature and feature manifold alignment for robust finetuning of vision-language models. *arXiv preprint arXiv:2603.27139*, 2026.

Sihui Dai, Christian Cianfarani, Arjun Bhagoji, Vikash Sehwag, and Prateek Mittal. Adapting to evolving adversaries with regularized continual robust training. *arXiv preprint arXiv:2502.04248*, 2025.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Sven Gowal, Krishnamurthy Dj Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. Scalable verified training for provably robust image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4842–4851, 2019.

Jiangpeng He, Zhihao Duan, and Fengqing Zhu. Cl-lora: Continual low-rank adaptation for rehearsal-free class-incremental learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pp. 30534–30544. Computer Vision Foundation / IEEE, 2025a. doi: 10.1109/CVPR52734.2025.02843.

Linpu He, Yanan Li, Bingze Li, Elvis Han Cui, and Donghui Wang. Dss-prompt: Dynamic-static synergistic prompting for few-shot class-incremental learning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 2703–2712, 2025b.

Michael Hersche, Geethan Karunaratne, Giovanni Cherubini, Luca Benini, Abu Sebastian, and Abbas Rahimi. Constrained few-shot class-incremental learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9047–9057. IEEE, 2022.

Andong Hua, Jindong Gu, Zhiyu Xue, Nicholas Carlini, Eric Wong, and Yao Qin. Initialization matters for adversarial transfer learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24831–24840. IEEE, 2024.

Xiaojun Jia, Yong Zhang, Baoyuan Wu, Ke Ma, Jue Wang, and Xiaochun Cao. Las-at: adversarial training with learnable attack strategy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13398–13408, 2022.

Haeyong Kang, Rusty John Lloyd Mina, Sultan Rizky Hikmawan Madjid, Jaehong Yoon, Mark Hasegawa-Johnson, Sung Ju Hwang, and Chang D Yoo. Forget-free continual learning with winning subnetworks. In *International Conference on Machine Learning*, pp. 10734–10750. PMLR, 2022.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 09–15 Jun 2019.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Inf. Proc. Systems (NIPS)*, pp. 1097–1105, 2012.

Alexey Kurakin, Ian J Goodfellow, Samy Bengio, et al. Adversarial examples in the physical world. *Artificial intelligence safety and security*, 2, 2018.

Lin Li, Haoyan Guan, Jianing Qiu, and Michael Spratling. One prompt word is enough to boost adversarial robustness for pre-trained vision-language models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24408–24419. IEEE, 2024.

Z. Li and D. Hoiem. Learning without forgetting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017.

Ye Liu and Meng Yang. Sec-prompt: Semantic complementary prompting for few-shot class-incremental learning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25643–25656. IEEE, 2025.

Mao-Lin Luo, Zi-Hao Zhou, Yi-Lin Zhang, Yuanyu Wan, Min-Ling Zhang, and Tong Wei. Keeplora: Continual learning with residual gradient adaptation. In *International Conference on Learning Representations*, volume 2026, pp. 136017–136040, 2026.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Mark D McDonnell, Dong Gong, Amin Parvaneh, Ehsan Abbasnejad, and Anton van den Hengel. Ranpac: Random projections and pre-trained models for continual learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Xiaoyue Mi, Fan Tang, Zonghan Yang, Danding Wang, Juan Cao, Peng Li, and Yang Liu. Adversarial robust memory-based continual learner. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 562–572. IEEE, 2025.

Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. A self-supervised approach for adversarial robustness. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 259–268. IEEE, 2020.

G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019.

Keon-Hee Park, Kyungwoo Song, and Gyeong-Moon Park. Pre-trained vision and language transformers are few-shot incremental learners. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23881–23890. IEEE, 2024.

Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *International conference on machine learning*, pp. 8093–8104. PMLR, 2020.

Ernest K Ryu and Stephen Boyd. Primer on monotone operator methods. *Appl. comput. math*, 15 (1):3–43, 2016.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. *Advances in neural information processing systems*, 31, 2018.

Jinxin Shi, Jiabao Zhao, Yifan Yang, Xingjiao Wu, Jiawen Li, and Liang He. Lark: Low-rank updates after knowledge localization for few-shot class-incremental learning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3607–3617. IEEE, 2025.

Maurice Sion. On general minimax theorems. *Pacific Jouronal of Mathematics*, 8(1):171–176, 1958.

Zeyin Song, Yifan Zhao, Yujun Shi, Peixi Peng, Li Yuan, and Yonghong Tian. Learning with fantasy: Semantic-aware virtual contrastive constraint for few-shot class-incremental learning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24183–24192. IEEE, 2023.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Licheng Tang, Yiyang Cai, Jiaming Liu, Zhibin Hong, Mingming Gong, Minhu Fan, Junyu Han, Jingtuo Liu, Errui Ding, and Jingdong Wang. Few-shot font generation by learning fine-grained local styles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7895–7904, 2022.

Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12180–12189. IEEE, 2020.

John von Neumann. Zur theorie der gesellschaftsspiele. volume 100, pp. 295–320. 1928. doi: 10.1007/BF01448847.

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. volume 36, pp. 15060–15076, 2023.

Qinzhe Wang, Zixuan Chen, Keke Huang, Xiu Su, Chunhua Yang, and Chang Xu. Consistency-driven calibration and matching for few-shot class incremental learning. In *International Conference on Learning Representations*, volume 2026, pp. 1255–1279, 2026a.

Sibo Wang, Jie Zhang, Zheng Yuan, and Shiguang Shan. Pre-trained model guided fine-tuning for zero-shot adversarial robustness. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24502–24511. IEEE, 2024.

Wenxuan Wang, Chenglei Wang, Chengzhi Yan, Xuelin Qian, and Yanning Zhang. Robustness under data scarcity: Few-shot continual adversarial training for evolving threats. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 34908–34917, 2026b.

Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International conference on learning representations*, 2019.

Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial effects through randomization. *arXiv preprint arXiv:1711.01991*, 2017.

Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L Yuille, and Quoc V Le. Adversarial examples improve image recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 816–825. IEEE, 2020.

Wufei Xie, Yalin Wang, Chenliang Liu, Zhaohui Jiang, and Xue Yang. Flexi-fscil: Adaptive knowledge retention for breaking the stability-plasticity dilemma in few-shot class-incremental learning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2451–2460. IEEE, 2025.

Fei Ye, Yulong Zhao, Qihe Liu, Junlin Chen, Adrian G Bors, Rongyao Hu, et al. Dynamic siamese expansion framework for improving robustness in online continual learning. volume 38, pp. 71293–71312, 2026.

Jia-Li Yin, Weijian Wang, Wei Lin, Ximeng Liu, et al. Adversarial-inspired backdoor defense via bridging backdoor and adversarial attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 9508–9516, 2025.

Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12450–12459. IEEE, 2021.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pp. 7472–7482. PMLR, 2019.

Linglan Zhao, Jing Lu, Yunlu Xu, Zhanzhan Cheng, Dashan Guo, Yi Niu, and Xiangzhong Fang. Few-shot class-incremental learning via class-aware bilateral distillation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11838–11847. IEEE, 2023.

Haizhong Zheng, Ziqi Zhang, Juncheng Gu, Honglak Lee, and Atul Prakash. Efficient adversarial training with transferable adversarial examples. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1178–1187. IEEE, 2020.

Qinghao Zhong, Bingzhi Chen, Yishu Liu, Minhua Lu, and Guangming Lu. Remoe: Region-mixture experts for adversarially-robust vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 37674–37683, 2026.

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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9036–9046. IEEE, 2022.

Yuhang Zhou and Zhongyun Hua. Defense without forgetting: Continual adversarial defense with anisotropic & isotropic pseudo replay. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24263–24272, 2024.

Huiping Zhuang, Zhenyu Weng, Run He, Zhiping Lin, and Ziqian Zeng. Gkeal: Gaussian kernel embedded analytic learning for few-shot class incremental task. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7746–7755. IEEE, 2023.

A Detailed Review of Related Work

In this section, we discuss prior work closely related to our study. We first examine few-shot class-incremental learning through decision-space adaptation and representation-space enhancement. Subsequently, we review adversarially robust continual learning from stationary robustness objectives to robustness preservation under evolving threats and sequential class-space changes.

A.1 Few-Shot Class-Incremental Learning.

FSCIL expands the label space using only a few examples per novel class while preserving previously learned categories. TOPIC Tao et al. (2020) modeled category knowledge with a neural-gas graph, expanding it for novel classes while preserving prior topology. Subsequent methods can be broadly viewed according to whether incremental adaptation is primarily performed in the decision space or in the representation space. Decision-space methods retain the learned representation and adapt session-wise classifiers or prototypes Zhang et al. (2021); Hersche et al. (2022); Zhuang et al. (2023); Wang et al. (2023); Chen et al. (2025), whereas representation-space methods reshape feature geometry to improve class separability and cross-session consistency Ahmed et al. (2024); Song et al. (2023); Zhao et al. (2023); Park et al. (2024); Liu & Yang (2025); Shi et al. (2025). A complementary line jointly optimize both spaces, coupling decision calibration with feature adaptation to improve classifier reliability Zhou et al. (2022); Wang et al. (2026a); Xie et al. (2025).

Decision-Space Adaptation on Frozen Representations. With the visual encoder held fixed, this line of work incorporates novel categories through class-level decision adaptation over frozen features. CEC Zhang et al. (2021) decouples representation learning from classifier evolution, using a graph trained on pseudo-incremental tasks to propagate context among session classifiers. C-FSCIL Hersche et al. (2022) emphasizes stability with a frozen meta-learned extractor, a trainable fixed-size layer, and growing class-vector memory. Its quasi-orthogonal vectors reduce interference as classes accumulate. GKEAL Zhuang et al. (2023) formulates FSCIL as recursive analytic learning, combining a Gaussian-kernel analytic module with augmented feature concatenation. Its weight-invariant recursion incorporates successive sessions without iterative optimization. TEEN Wang et al. (2023) corrects novel-prototype bias through training-free fusion with weighted base prototypes, reducing the mismatch between base and incremental classes. BiMC Chen et al. (2025) extends calibration to pre-trained vision-language models through intra- and inter-modal calibration, adjusting visual and textual representations for more reliable decisions. Collectively, these methods confine incremental adaptation largely to class-level decision making, thereby reducing parameter updates and representation drift while limiting their capacity to reshape the underlying cross-session feature geometry.

Representation-Space Adaptation for Cross-Session Separability. Under few-shot supervision, these methods improve incremental recognition by enforcing discriminative and session-consistent feature structures. Contrastive and geometric objectives directly shape the embedding space. OrCo Ahmed et al. (2024) combines supervised and self-supervised contrastive learning with feature perturbation and orthogonality constraints to enlarge inter-class margins and reserve space for future classes, while SAVC Song et al. (2023) uses transformation-generated virtual classes in a contrastive fantasy space to separate base and novel classes. Distillation and prompt-based methods transfer or inject knowledge across sessions. CABD Zhao et al. (2023) distills complementary knowledge from the base and preceding-session models using class-wise semantic similarity, whereas PriViLege Park et al. (2024) adapts pre-trained vision-language transformers through prompting, pre-trained knowledge tuning, entropy-based divergence, and semantic knowledge distillation. SEC-Prompt Liu & Yang (2025) learns complementary D-Prompt and ND-Prompt modules, using ND-Prompt for data augmentation and Prompt Clustering Loss to stabilize D-Prompt learning. Lark Shi et al. (2025) provides a parameter-efficient alternative by localizing knowledge-sensitive parameters with the Fisher information matrix and applying rank-one updates. Other methods couple feature adaptation with decision calibration, FACT Zhou et al. (2022) reserves embedding space for future classes through virtual prototypes. ConCM Wang et al. (2026a) couples memory-aware prototype calibration with dynamic structure matching. Flexi-FSCIL Xie et al. (2025) fuses frozen and trainable representations through hybrid distillation and prototype-offset correction. Collectively, these methods strengthen cross-session discrimination through geometric, knowledge-transfer, and parameter-efficient mechanisms, but are largely evaluated on clean inputs, leaving their reliability under adversarial perturbations insufficiently examined.

A.2 Adversarially robust continual learning

Early adversarial robustness research predominantly assumes a fixed data distribution, label space, and perturbation constraint. Foundational studies first established how to balance natural accuracy and robust error, TRADES Zhang et al. (2019) formulates robustness as a principled trade-off, after which IBP Gowal et al. (2019), NRP Naseer et al. (2020), Adv-SS-Pretraining Chen et al. (2020), and BPFC Addepalli et al. (2020) explored verifiable training and robust representation construction. Another predominant line focuses on broadening the scope of robustness optimization Zheng et al.

(2020); Xie et al. (2020); Jia et al. (2022). Subsequent work studies robustness during transfer and adaptation through robust initialization Hua et al. (2024), guidance from pre-trained models Wang et al. (2024), prompt tuning Li et al. (2024), geometric regularization Chopra et al. (2026), and expert routing Zhong et al. (2026). Despite these advances, existing methods are primarily developed for a single training regime and do not address the preservation of previously acquired robustness under sequential model updates.

Continual adversarial learning removes the stationary assumption by studying how models retain previously acquired robustness throughout sequential adaptation. LiDER Bonicelli et al. (2022) mitigates instability caused by repeated rehearsal by imposing layer-wise Lipschitz constraints on replay examples, thereby promoting smoother decision boundaries. AIR Zhou & Hua (2024) addresses continual defense under changing attack conditions through anisotropic and isotropic pseudo-replay together with output regularization. FLAIR Cho et al. (2025) formulates adversarially robust class-incremental learning and identifies the flat input-space loss landscape of robust models as a property disrupted by incremental updates, motivating its preservation during adversarial adaptation. ARMCL Mi et al. (2025) combines gradient-based memory selection with logit calibration to retain informative and robust exemplars. In online continual learning, DSEF Ye et al. (2026) employs static and dynamic Siamese components to construct lightweight task experts while preserving historical representations through Robust Dynamic Representation Optimization and Robust Feature Fusion. Dai et al. (2025) addresses evolving adversaries by regularizing the logit distance between clean and attacked inputs, thereby mitigating robustness degradation on previously encountered attacks. Under severe data scarcity, FS-CAT Wang et al. (2026b) studies continual adversarial training under evolving threats while maintaining a common label space rather than introducing new classes. Overall, existing adversarial continual-learning research has mainly considered evolving attacks or conventional class and task incremental updates, whereas few-shot settings generally retain a shared label space. The joint problem of adversarial robustness and few-shot class expansion remains insufficiently explored.

B THE PROOFS FOR THEORETICAL RESULTS

Proof of Theorem 3.1. Since $\mathcal{A}$ and $\mathcal{D}$ are compact and convex and u is continuous on $\mathcal{A}\times\mathcal{D}$, convex in Θ_t, and concave in $\tilde{\Theta}_t$, the game $\mathcal{G}$ is a finite-dimensional convex-concave game. By the Minimax Theorem von Neumann (1928), we have:

$$\min_{\Theta_t\in\mathcal{D}} \max_{\tilde{\Theta}_t\in\mathcal{A}} u(\tilde{\Theta}_t,\Theta_t) \;=\; \max_{\tilde{\Theta}_t\in\mathcal{A}} \min_{\Theta_t\in\mathcal{D}} u(\tilde{\Theta}_t,\Theta_t). \tag{16}$$

The existence of a saddle point $(\tilde{\Theta}_t^\star,\Theta_t^\star)$ satisfying Eq. (15) then follows directly from Sion's Minimax Theorem Sion (1958), which guarantees a saddle point whenever u is quasiconvex-quasiconcave on compact convex sets. This saddle point constitutes the Nash Equilibrium of the zero-sum game $\mathcal{G}$.

Convergence of Alternating Gradient Updates. The two-stage alternating update corresponds to gradient *ascent* on $\tilde{\Theta}_t$ (attacker maximizes u) and gradient *descent* on Θ_t (defender minimizes $\mathcal{L}_{\text{def}}$). To unify both directions, define the joint iterate $\mathbf{z}^{(k)}=(\tilde{\Theta}_t^{(k)},\Theta_t^{(k)})^\top$ and the joint operator:

$$\mathbf{F}(\mathbf{z}) \;=\; \begin{pmatrix} -\nabla_{\tilde{\Theta}_t} u(\tilde{\Theta}_t,\Theta_t) \\ +\nabla_{\Theta_t} u(\tilde{\Theta}_t,\Theta_t) \end{pmatrix}, \tag{17}$$

so that the unified step $\mathbf{z}^{(k+1)}=\mathbf{z}^{(k)}-\eta\,\mathbf{F}(\mathbf{z}^{(k)})$ simultaneously implements gradient ascent on $\tilde{\Theta}_t$ and gradient descent on Θ_t.

Proposition B.1 (Convergence of Alternating Updates) *Let u be L-smooth and μ-strongly-convex-concave, with condition number $\kappa=L/\mu\geq 1$. Under learning rate $\eta=\frac{\mu}{L^2}$, the alternating gradient updates:*

$$\tilde{\Theta}_t^{(k+1)} = \tilde{\Theta}_t^{(k)} + \eta\,\nabla_{\tilde{\Theta}_t}\mathcal{L}_{\text{CE}}\Big(\mathcal{E}_t,\, F_g(\mathbf{x},\tilde{\mathcal{E}}_t^{(k)}),\, y\Big), \tag{18}$$

$$\Theta_t^{(k+1)} = \Theta_t^{(k)} - \eta\,\nabla_{\Theta_t}\mathcal{L}_{\text{def}}^{(k)}, \tag{19}$$

converge to the Nash Equilibrium $(\tilde{\Theta}_t^\star,\Theta_t^\star)$ at a linear rate:

$$\left\| \begin{pmatrix} \tilde{\Theta}_t^{(k)} \\ \Theta_t^{(k)} \end{pmatrix} - \begin{pmatrix} \tilde{\Theta}_t^\star \\ \Theta_t^\star \end{pmatrix} \right\|_2 \;\leq\; \left(1-\frac{\mu^2}{L^2}\right)^{k/2} \left\| \begin{pmatrix} \tilde{\Theta}_t^{(0)} \\ \Theta_t^{(0)} \end{pmatrix} - \begin{pmatrix} \tilde{\Theta}_t^\star \\ \Theta_t^\star \end{pmatrix} \right\|_2. \tag{20}$$

The detailed proof is provided in **Appendix** B.

Remark. The linear convergence rate is governed by the contraction factor $\rho^\star = 1 - \mu^2/L^2 = 1 - 1/\kappa^2$, where $\kappa = L/\mu$ is the condition number of the payoff u. A smaller κ (i.e., a better-conditioned game) accelerates convergence toward the Nash Equilibrium. The margin-based CKA regularization $\mathcal{L}_{\mathrm{CKA}}$ in $\mathcal{L}_{\mathrm{def}}$ acts as a structured regularizer that improves the local curvature of the defender's loss landscape by suppressing degenerate directions arising from representational redundancy between $\mathcal{E}_0$ and $\mathcal{E}_t$, thereby reducing κ and accelerating convergence in practice.

Proof of Proposition 3.2. Let $\mathbf{z}^\star = (\tilde{\Theta}_t^\star, \Theta_t^\star)^\top$ denote the Nash Equilibrium guaranteed by Theorem 3.1, and let $\mathbf{F}$ be the joint operator defined in Eq. (17). Under μ-strong-convex-concavity of u, $\mathbf{F}$ is μ-strongly monotone Ryu & Boyd (2016):

$$\langle \mathbf{F}(\mathbf{z}) - \mathbf{F}(\mathbf{z}^\star),\, \mathbf{z} - \mathbf{z}^\star \rangle \;\geq\; \mu \left\| \mathbf{z} - \mathbf{z}^\star \right\|_2^2. \tag{21}$$

Under L-smoothness, $\mathbf{F}$ is L-Lipschitz continuous:

$$\left\| \mathbf{F}(\mathbf{z}) - \mathbf{F}(\mathbf{z}^\star) \right\|_2 \;\leq\; L \left\| \mathbf{z} - \mathbf{z}^\star \right\|_2. \tag{22}$$

One step of the unified descent $\mathbf{z}^{(k+1)} = \mathbf{z}^{(k)} - \eta\, \mathbf{F}(\mathbf{z}^{(k)})$ gives:

$$\begin{aligned} \left\| \mathbf{z}^{(k+1)} - \mathbf{z}^\star \right\|_2^2 &= \left\| \mathbf{z}^{(k)} - \mathbf{z}^\star - \eta\, \mathbf{F}(\mathbf{z}^{(k)}) \right\|_2^2 \\ &= \left\| \mathbf{z}^{(k)} - \mathbf{z}^\star \right\|_2^2 - 2\eta \left\langle \mathbf{F}(\mathbf{z}^{(k)}) - \mathbf{F}(\mathbf{z}^\star),\, \mathbf{z}^{(k)} - \mathbf{z}^\star \right\rangle + \eta^2 \left\| \mathbf{F}(\mathbf{z}^{(k)}) - \mathbf{F}(\mathbf{z}^\star) \right\|_2^2, \end{aligned} \tag{23}$$

where we used $\mathbf{F}(\mathbf{z}^\star) = \mathbf{0}$ in the first inner-product term and kept it for clarity. Substituting Eq. (21) and Eq. (22) into Eq. (23):

$$\begin{aligned} \left\| \mathbf{z}^{(k+1)} - \mathbf{z}^\star \right\|_2^2 &\leq \left\| \mathbf{z}^{(k)} - \mathbf{z}^\star \right\|_2^2 - 2\eta\mu \left\| \mathbf{z}^{(k)} - \mathbf{z}^\star \right\|_2^2 + \eta^2 L^2 \left\| \mathbf{z}^{(k)} - \mathbf{z}^\star \right\|_2^2 \\ &= \underbrace{\left(1 - 2\eta\mu + \eta^2 L^2\right)}_{\rho(\eta)} \left\| \mathbf{z}^{(k)} - \mathbf{z}^\star \right\|_2^2. \end{aligned} \tag{24}$$

The contraction factor $\rho(\eta) = 1 - 2\eta\mu + \eta^2 L^2$ is a convex quadratic in η, minimized at:

$$\eta^\star = \frac{\mu}{L^2}, \qquad \rho(\eta^\star) = 1 - 2 \cdot \frac{\mu}{L^2} \cdot \mu + \frac{\mu^2}{L^4} \cdot L^2 = 1 - \frac{\mu^2}{L^2} = 1 - \frac{1}{\kappa^2} \;\in\; (0, 1), \tag{25}$$

where $\kappa = L/\mu \geq 1$, so $\rho(\eta^\star) < 1$ and the iteration is contractive. Applying Eq. (24) recursively with $\eta = \eta^\star$:

$$\left\| \mathbf{z}^{(k)} - \mathbf{z}^\star \right\|_2^2 \;\leq\; \left(1 - \frac{\mu^2}{L^2}\right)^k \left\| \mathbf{z}^{(0)} - \mathbf{z}^\star \right\|_2^2. \tag{26}$$

Taking square roots on both sides yields Eq. (20).

C Experimental Details

Experimental setup. We evaluate FSCIL-AT on CIFAR-100, mini-ImageNet, CUB-200, and ImageNet-R following standard FSCIL protocols He et al. (2025b). CIFAR-100 and mini-ImageNet use 60 base classes followed by eight 5-way 5-shot sessions, while CUB-200 and ImageNet-R use 100 base classes followed by ten 10-way 5-shot sessions. We report session-averaged clean Top-1 accuracy and final-session robust Top-1 accuracy over all learned classes against FGSM Goodfellow et al. (2014), ℓ_∞-PGD Madry et al. (2017), ℓ_2-PGD Madry et al. (2017), BIM Kurakin et al. (2018), and CW Carlini & Wagner (2017). *Avg-All* denotes the unweighted mean of the clean score and the five attack-specific scores. Results are averaged over three independent runs. All methods use ViT-B/16, and trainable methods use 30 base-session epochs and 10 epochs per incremental session.

Compared methods. We compare with general continual learners LwF Li & Hoiem (2017) and RanPAC McDonnell et al. (2024); parameter-efficient methods CL-LoRA He et al. (2025a) and KeepLoRA Luo et al. (2026); FSCIL methods TEEN Wang et al. (2023), BiMC Chen et al. (2025), and ConCM Wang et al. (2026a); and robust continual learners AIR Zhou & Hua (2024), DSEF Ye et al. (2026), and ARMCL Mi et al. (2025). Methods without native adversarial optimization are evaluated under standard training (*Std*) and a shared FGSM adversarial-training variant (*Adv*), while robust methods retain their original objectives (*Native*). All applicable methods use the same ViT-B/16 backbone and FSCIL class splits. Trainable methods use 30 epochs for the base session and

10 epochs for each incremental session, while BiMC retains its method-specific CLIP ViT-B/16 encoder.

Implementation details. All inputs are resized to 224×224. We initialize the frozen ViT-B/16 backbone with weights pretrained on ImageNet-21K and fine-tuned on ImageNet-1K. The LoRA rank is set to 16 for the query, key, value, and output projections, and the base-expert gates are initialized to 0.5. CKA regularization is applied to Transformer blocks 6–11. The learnable parameters are optimized using SGD with an initial learning rate of 0.03, momentum 0.9, weight decay 10^{-4}, batch size 64, and cosine learning-rate decay. We train the base session for 30 epochs and each incremental session for 10 epochs. Hyperparameter settings and sensitivity analyses for all modules are provided in **Appendix G**.

Adversarial settings. FGSM training uses an ℓ_∞ radius of $8/255$, a step size of $10/255$. Adversarial updates occur every three base-session mini-batches to balance clean learning, robustness, and cost, but every incremental mini-batch to ensure sufficient robust supervision from scarce data. To stabilize clean adaptation before adversarial optimization, we use a clean warm-up that trains only on clean samples for the first 15 base-session and 3 incremental-session epochs. For robustness evaluation, we follow the adversarial attack configurations of DSEF Ye et al. (2026). FGSM uses $\epsilon = 8/255$, ℓ_∞-PGD and BIM use $\epsilon = 8/255$, a step size of $2/255$, and 10 iterations, ℓ_2-PGD uses $\epsilon = 128/255$, a step size of $15/255$, and 10 iterations, CW uses $c = 1$, $\kappa = 0$, 100 iterations, and a learning rate of 0.01.

Evaluation metrics. We define session-averaged clean Top-1 accuracy, final-session robust Top-1 accuracy, and *Avg-All* to evaluate clean and adversarial performance under the FSCIL protocol. Let $\mathcal{Q}_t^{\text{clean}}$ denote the clean test set containing all classes in $\mathcal{C}_{0:t}$, and let $\mathcal{Q}_{T,a}^{\text{adv}}$ denote its adversarial counterpart generated by attack a after the final session. The clean and attack-specific robust scores are defined as

$$A_{\text{Clean}} = \frac{1}{T+1} \sum_{t=0}^{T} \text{Acc}\big(\mathcal{Q}_t^{\text{clean}}\big), \qquad A_{\text{Rob}}^{a} = \text{Acc}\big(\mathcal{Q}_{T,a}^{\text{adv}}\big), \tag{27}$$

where $\text{Acc}(\cdot)$ denotes Top-1 accuracy and $a \in \{\text{FGSM}, \ell_\infty\text{-PGD}, \ell_2\text{-PGD}, \text{BIM}, \text{CW}\}$. The overall score is

$$\text{Avg-All} = \frac{1}{6}\left(A_{\text{Clean}} + \sum_{a} A_{\text{Rob}}^{a}\right), \tag{28}$$

where the summation covers the five attacks above.

D Further Analysis of Adversarial Overfitting and Rival Coverage

D.1 Analysis Adversarial Overfitting under Data Scarcity

Adversarial robustness generally requires substantially more training data than standard generalization, and adversarial training itself can exhibit pronounced overfitting behavior Schmidt et al. (2018); Rice et al. (2020). To examine whether the robustness learned from limited training samples generalizes to unseen test samples, we further investigate the performance of adversarial training in FSCIL under different shot settings. Following the CUB-200 protocol described in Appendix C, we vary the number of training samples per novel class as $K \in \{1, 3, 5, 10, 20\}$ and compare Naive AT with ATPGO. All other training and evaluation settings are held fixed to isolate the effect of ATPGO.

After each incremental session S_t, we evaluate the resulting model on the novel classes introduced in that session. For each attack, adversarial examples are generated independently from the training and test samples of C_t. We consider $\mathcal{A}_{\text{eval}} = \{\text{FGSM}, \ell_\infty\text{-PGD}, \ell_2\text{-PGD}, \text{BIM}, \text{CW}\}$. For method $m \in \{\text{Naive AT}, \text{ATPGO}\}$, let $A_{m,t,a}^{\text{train}}(K)$ and $A_{m,t,a}^{\text{test}}(K)$ denote the mean per-class Top-1 accuracies of the model after session S_t under attack $a \in \mathcal{A}_{\text{eval}}$. Specifically, $A_{m,t,a}^{\text{train}}(K)$ is evaluated on adversarial examples generated from the training samples of the novel classes introduced in S_t, whereas $A_{m,t,a}^{\text{test}}(K)$ is evaluated on adversarial examples generated from the corresponding test samples. Each accuracy is computed separately for every novel class in C_t and then averaged across these classes. The corresponding session-wise robust generalization gap is defined as $G_{m,t,a}(K) = A_{m,t,a}^{\text{train}}(K) - A_{m,t,a}^{\text{test}}(K)$. We aggregate the two robust accuracies over all incremental

sessions and evaluation attacks:

$$\overline{A}_m^{\text{train}}(K) = \frac{1}{T\,|\mathcal{A}_{\text{eval}}|} \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_{\text{eval}}} A_{m,t,a}^{\text{train}}(K). \tag{29}$$

$$\overline{A}_m^{\text{test}}(K) = \frac{1}{T\,|\mathcal{A}_{\text{eval}}|} \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_{\text{eval}}} A_{m,t,a}^{\text{test}}(K). \tag{30}$$

where $T = 10$ for CUB-200. Accordingly, the robust generalization gap reported in Fig. 1 (a) is

$$\overline{G}_m(K) = \frac{1}{T\,|\mathcal{A}_{\text{eval}}|} \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_{\text{eval}}} G_{m,t,a}(K) = \overline{A}_m^{\text{train}}(K) - \overline{A}_m^{\text{test}}(K). \tag{31}$$

Fig. 1 (b) reports $\overline{A}_m^{\text{test}}(K)$ as the session-end robust accuracy on test samples. The intervals shown in the figure represent the variation estimated through repeated resampling.

Fig. 1 (a) and (b) reveal that, at small K, Naive AT attains substantially higher robustness on the training samples than on the test samples, indicating pronounced adversarial overfitting. As K increases, the robust generalization gap narrows and the session-end test robustness improves, confirming that this phenomenon is exacerbated by limited training data. Under identical experimental settings, ATPGO consistently achieves a smaller robust generalization gap and higher test robustness across all evaluated values of K, demonstrating its effectiveness in alleviating adversarial overfitting. This improvement is particularly pronounced in the low-shot regime, where ATPGO diversifies adversarial supervision by exposing the model to a broader range of adversarial variations, thereby mitigating overfitting to the limited attack patterns induced by scarce training samples.

D.2 Analysis of Rival-Class Coverage

To further examine the diversity of adversarial supervision under data scarcity, we compare the rival-class coverage of Naive AT and ATPGO under the same CUB-200 shot settings. Each novel class is treated as a source class, and its coverage is measured by the number of distinct rival classes exposed by adversarial examples generated from its training samples.

For an adversarial example x^{adv} with ground-truth label y, its rival class is defined as the class with the highest predicted probability after excluding y:

$$y^{\text{riv}} = \arg\max_{c \neq y} \hat{p}_c(x^{\text{adv}}), \tag{32}$$

where $\hat{p}_c(x^{\text{adv}})$ denotes the predicted probability of class c. For Naive AT, we use the predictions of its trained model; for ATPGO, we use the predictions of the defender expert for both self-generated and transferred adversarial examples. We collect the rival classes associated with each source class and count the distinct classes to quantify the diversity of exposed rival-class directions. For each K and each source class, rival-class coverage is evaluated under a fixed budget of 100 adversarial examples per method. Naive AT generates all 100 adversarial examples using its own model. ATPGO uses 50 self-generated adversarial examples from the defender expert and 50 transferred adversarial examples generated by the independently optimized attacker expert. The adversarial example budget is allocated uniformly across the K training samples of each source class. Adversarial examples are generated using the FGSM configuration specified in Appendix C. The equal adversarial example budget across the two methods controls for the effect of sample quantity on rival-class coverage.

For a source class y at a given K, let $\mathcal{R}_{\text{N}}(y; K)$ denote the set of distinct rival classes exposed by Naive AT. For ATPGO, let $\mathcal{R}_{\text{self}}(y; K)$ and $\mathcal{R}_{\text{trans}}(y; K)$ denote the rival-class sets obtained from self-generated and transferred adversarial examples, respectively. Its overall rival-class set is

$$\mathcal{R}_{\text{A}}(y; K) = \mathcal{R}_{\text{self}}(y; K) \cup \mathcal{R}_{\text{trans}}(y; K). \tag{33}$$

Naive AT coverage is given by $|\mathcal{R}_{\text{N}}(y; K)|$. ATPGO coverage is decomposed into rival classes shared with Naive AT and those exclusively exposed by ATPGO:

$$|\mathcal{R}_{\text{A}}(y; K)| = |\mathcal{R}_{\text{A}}(y; K) \cap \mathcal{R}_{\text{N}}(y; K)| + |\mathcal{R}_{\text{A}}(y; K) \setminus \mathcal{R}_{\text{N}}(y; K)|. \tag{34}$$

The two terms correspond to "ATPGO Shared", which counts rival classes exposed by both Naive AT and ATPGO, and "ATPGO Exclusive", which counts rival classes exposed by ATPGO but absent

from the corresponding Naive AT set, as shown in Fig. 1 (c). All three reported quantities are averaged over the novel classes introduced across the incremental sessions.

Fig. 1 (c) shows that rival-class coverage increases with K for both methods, indicating that additional training samples expand the range of competing class directions exposed by adversarial examples. Under a fixed adversarial example budget, the smaller rival-class set of Naive AT indicates greater repetition of already exposed rivals. Although ATPGO allocates only half of this budget to self-generated adversarial examples, the shared and exclusive components show that its combined supervision retains substantial overlap with Naive AT's rival-class set while exposing additional rival classes at every evaluated K.

Table 4: **Clean and adversarial accuracy on CIFAR-100.** Top-1 accuracy (%) is reported. Avg-All is the unweighted mean over clean accuracy and the five adversarial attacks.

Method	Train	Clean	Robust accuracy					Avg-All
			FGSM	ℓ_∞-PGD	ℓ_2-PGD	BIM	CW	
LwF (ECCV'16)	Std	70.11±00.28	23.38±00.05	02.29±00.11	22.13±00.01	01.55±00.09	00.00±00.00	19.91
	Adv	69.68±00.06	25.32±00.34	04.99±00.16	23.99±00.11	06.23±00.15	00.00±00.00	21.70
RanPAC (NeurIPS'23)	Std	72.43±00.03	22.39±00.12	00.02±00.02	24.82±00.27	00.01±00.01	06.36±00.24	21.01
	Adv	72.42±00.07	31.86±00.55	00.81±00.19	27.16±00.14	00.75±00.08	09.43±00.17	23.74
CL-LoRA (CVPR'25)	Std	**84.86±00.19**	22.63±01.28	00.06±00.03	31.33±00.43	00.08±00.01	12.60±00.45	25.26
	Adv	83.75±00.27	26.39±03.41	00.34±00.09	38.11±01.12	00.23±00.07	27.05±01.22	29.31
KeepLoRA (ICLR'26)	Std	80.62±00.27	35.27±00.24	03.38±00.28	36.36±00.10	04.04±00.48	12.53±00.15	28.70
	Adv	80.09±00.31	44.87±00.98	08.11±02.03	37.38±00.55	05.78±01.41	23.09±00.45	33.22
TEEN (NeurIPS'23)	Std	79.23±00.00	14.61±00.04	01.86±00.04	19.59±00.04	00.68±00.01	00.00±00.00	19.33
	Adv	79.23±00.00	14.61±00.04	01.88±00.06	19.59±00.04	00.68±00.01	00.00±00.00	19.33
BiMC (CVPR'25)	Std	72.97±00.00	15.71±00.01	00.00±00.00	04.03±00.03	00.00±00.00	03.67±00.03	16.06
	Adv	73.06±00.04	16.64±00.08	00.00±00.00	04.57±00.08	00.00±00.00	04.20±00.06	16.41
ConCM (ICLR'26)	Std	72.49±00.47	17.53±00.12	00.50±00.03	18.74±00.10	00.26±00.06	03.25±00.01	18.80
	Adv	71.25±00.25	16.86±00.01	00.46±00.05	18.76±00.39	00.25±00.13	03.07±00.00	18.44
AIR (CVPR'24)	Native	35.78±03.25	14.92±00.87	07.09±01.68	13.44±00.93	07.16±01.32	00.00±00.00	13.07
DSEF (NeurIPS'25)	Native	26.56±20.82	07.61±05.64	01.34±00.43	05.42±04.16	02.46±01.34	00.59±00.56	07.33
ARMCL (ICCV'25)	Native	60.50±00.43	18.98±02.75	06.28±01.25	17.54±02.63	07.45±01.33	00.00±00.00	18.46
FSCIL-AT (Ours)	**ATPGO**	81.83±00.28	**53.87±00.84**	**11.02±00.97**	**41.68±01.06**	**09.90±00.72**	**36.39±02.07**	**39.12**

E Additional Benchmark Comparisons

As shown in Tables 4, 5 and 6, FSCIL-AT exceeds the strongest baseline in Avg-All by 5.90%, 8.02%, and 2.39% across CIFAR-100, mini-ImageNet, and ImageNet-R. Although some methods achieve high clean accuracy, their robustness under adversarial attacks remains limited. For example, TEEN reaches 97.97% clean accuracy on mini-ImageNet but only 0.02% under CW, while BiMC attains 75.57% clean accuracy on ImageNet-R yet achieves nearly zero accuracy under ℓ_∞-PGD and BIM. This discrepancy attributed to their training-free calibration strategies, which preserve or refine class-level predictions without explicitly adapting the visual representation to adversarial perturbations. In contrast, CL-LoRA and KeepLoRA retain trainable feature pathways. CL-LoRA's shared and task-specific adapters sustain high clean accuracy, while KeepLoRA projects LoRA gradients away from pretrained principal and prior-task feature directions, limiting interference during adversarial adaptation and attaining higher accuracy under iterative ℓ_∞-PGD and BIM attacks. FSCIL-AT slightly trails in clean accuracy but leads under every attack. Gated reuse of the frozen base LoRA expert allow the new expert to adapt without altering old-class predictions, while co-evolved adversaries diversify robust supervision.

F Attack-Wise Component Ablation

Fig. 6 compares three configurations under clean inputs and five white-box attacks. The w/o ATPGO setting retains MBCKAR but removes adversarial training, whereas w/o MBCKAR retains ATPGO without the CKA constraint. The full model ranks first under all attacks on both datasets. The higher clean accuracy of w/o ATPGO reflects the stronger optimization constraints imposed by ATPGO, which slightly reduce the emphasis on fitting scarce clean samples while substantially improving robustness through broader adversarial supervision. Relative to w/o ATPGO, the full model improves Rob-Avg by 16.92% and 20.24%, and Avg-All by 13.58% and 16.48% on CIFAR-100 and mini-ImageNet, respectively. FGSM, ℓ_∞-PGD, and BIM share the same ℓ_∞ perturbation geometry, with FGSM using a single update and ℓ_∞-PGD and BIM iteratively refining the perturbation. The marked

Table 5: **Clean and adversarial accuracy on mini-ImageNet.** Top-1 accuracy (%) is reported. Avg-All is the unweighted mean over clean accuracy and the five adversarial attacks.

Method	Train	Clean	Robust accuracy					Avg-All
			FGSM	ℓ_∞-PGD	ℓ_2-PGD	BIM	CW	
LwF (ECCV'16)	Std	91.85±00.61	52.52±00.01	19.56±00.30	51.42±00.33	17.67±00.44	00.06±00.04	38.85
	Adv	86.78±00.22	53.31±00.33	07.85±00.38	55.08±00.80	06.02±00.21	00.11±00.03	34.86
RanPAC (NeurIPS'23)	Std	76.13±00.03	40.48±00.20	00.38±00.04	47.58±00.25	00.21±00.07	23.33±00.39	31.35
	Adv	76.13±00.01	49.93±00.15	02.45±00.09	50.27±00.10	02.21±00.19	35.00±00.15	36.00
CL-LoRA (CVPR'25)	Std	94.99±00.02	54.21±00.44	01.90±00.19	66.12±00.34	01.57±00.22	51.92±00.15	45.12
	Adv	94.86±00.12	65.32±01.56	02.25±00.70	72.94±00.36	01.64±00.47	70.86±00.95	51.31
KeepLoRA (ICLR'26)	Std	91.29±00.30	55.44±00.30	09.33±02.72	60.58±00.78	08.03±02.39	36.16±00.82	43.47
	Adv	90.56±00.07	65.45±00.29	15.58±02.57	69.95±00.49	14.08±02.68	53.03±01.11	51.44
TEEN (NeurIPS'23)	Std	**97.97±00.00**	64.17±00.01	17.27±00.11	53.67±00.02	17.01±00.01	00.02±00.00	41.69
	Adv	**97.97±00.00**	64.17±00.01	17.28±00.01	53.67±00.02	17.01±00.01	00.02±00.00	41.69
BiMC (CVPR'25)	Std	90.91±00.00	36.38±00.01	00.01±00.01	32.35±00.08	00.00±00.00	26.23±00.09	30.98
	Adv	90.95±00.02	37.98±00.06	00.02±00.01	34.24±00.09	00.00±00.00	27.16±00.10	31.73
ConCM (ICLR'26)	Std	93.35±00.04	51.06±00.29	04.64±00.11	54.95±00.77	03.24±00.10	19.17±00.66	37.74
	Adv	92.93±00.08	49.53±00.40	05.56±00.22	53.40±00.09	03.67±00.14	17.75±00.26	37.14
AIR (CVPR'24)	Native	62.99±05.03	26.07±07.62	16.81±06.13	27.57±07.58	17.27±05.56	00.01±00.01	25.12
DSEF (NeurIPS'25)	Native	21.11±31.51	06.88±08.52	01.89±00.82	05.08±05.28	02.26±01.30	01.49±01.22	06.45
ARMCL (ICCV'25)	Native	92.07±00.61	47.61±00.80	14.73±01.53	57.18±00.67	14.09±01.38	00.17±00.05	37.64
FSCIL-AT (Ours)	**ATPGO**	93.80±00.25	**71.76±00.07**	**24.29±00.81**	**73.46±00.94**	**22.32±00.11**	**71.13±00.09**	**59.46**

Table 6: **Clean and adversarial accuracy on ImageNet-R.** Top-1 accuracy (%) is reported. Avg-All is the unweighted mean over clean accuracy and the five adversarial attacks.

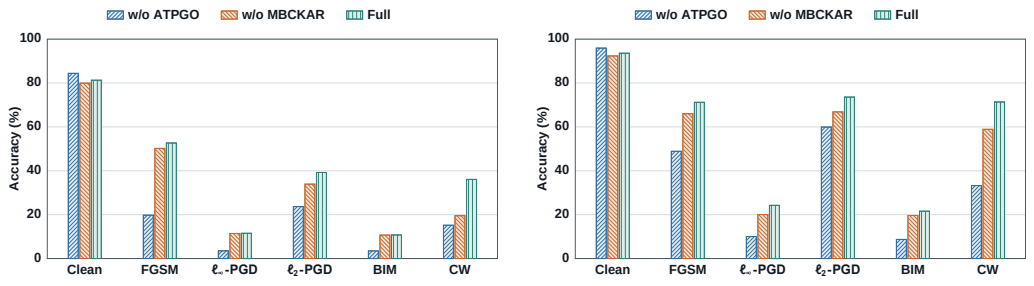
Method	Train	Clean	Robust accuracy					Avg-All
			FGSM	ℓ_∞-PGD	ℓ_2-PGD	BIM	CW	
LwF (ECCV'16)	Std	54.34±00.04	12.43±00.25	04.02±00.02	13.52±00.10	03.69±00.10	00.08±00.02	14.68
	Adv	55.51±00.17	11.35±00.22	00.46±00.02	14.31±00.08	00.37±00.00	00.15±00.03	13.69
RanPAC (NeurIPS'23)	Std	58.82±00.19	09.48±00.20	00.25±00.06	29.57±00.27	00.22±00.05	22.68±00.43	20.17
	Adv	58.72±00.20	21.00±00.08	03.40±00.14	32.41±00.39	03.24±00.03	27.53±00.64	24.38
CL-LoRA (CVPR'25)	Std	**79.45±00.08**	15.57±00.35	00.70±00.07	41.85±00.37	00.56±00.11	38.22±00.40	29.39
	Adv	79.03±00.12	34.63±00.96	03.43±00.43	46.96±00.79	03.41±00.36	43.71±01.05	35.20
KeepLoRA (ICLR'26)	Std	72.82±00.18	23.83±00.24	03.20±00.05	35.73±00.12	03.20±00.05	26.85±00.20	27.61
	Adv	72.66±00.18	34.62±01.12	04.70±00.50	42.66±00.76	06.28±00.19	36.97±00.35	32.98
TEEN (NeurIPS'23)	Std	61.14±00.00	10.52±00.00	00.66±00.04	20.66±00.08	00.54±00.01	00.38±00.00	15.65
	Adv	61.14±00.00	10.52±00.00	00.69±00.08	20.66±00.08	00.54±00.01	00.38±00.00	15.66
BiMC (CVPR'25)	Std	74.92±00.00	16.97±00.00	00.04±00.03	18.81±00.03	00.02±00.02	18.90±00.05	21.61
	Adv	75.57±00.12	17.46±00.07	00.03±00.00	19.44±00.11	00.01±00.01	19.13±00.05	21.94
ConCM (ICLR'26)	Std	59.37±00.01	09.88±00.00	00.12±00.00	21.53±00.08	00.14±00.01	11.38±00.35	17.07
	Adv	59.26±00.06	09.86±00.23	00.10±00.04	21.37±00.12	00.09±00.01	11.39±00.02	17.01
AIR (CVPR'24)	Native	24.16±01.37	04.65±00.34	02.65±00.21	04.94±00.54	02.77±00.43	00.02±00.01	06.53
DSEF (NeurIPS'25)	Native	11.23±00.33	00.81±00.38	00.85±00.39	00.81±00.38	00.81±00.38	00.78±00.37	02.55
ARMCL (ICCV'25)	Native	46.04±01.21	07.25±00.47	02.15±00.31	11.57±00.97	01.97±00.29	00.13±00.03	11.52
FSCIL-AT (Ours)	**ATPGO**	75.92±00.59	**37.21±01.26**	**06.93±01.13**	**49.04±00.93**	**06.73±00.85**	**49.71±00.76**	**37.59**

degradation after removing ATPGO shows that its co-evolving transferred and self-generated examples directly strengthen resistance to the local ℓ_∞ vulnerabilities encountered during training. The gains also extend to ℓ_2-PGD and CW because ATPGO explores changing decision-boundary weaknesses rather than repeatedly optimizing a fixed perturbation pattern: transferred examples provide attack directions that are not tied to one norm, while self-generated examples adapt supervision to the defender's current boundary.

G Anlysis of hyperparameter sensitivity

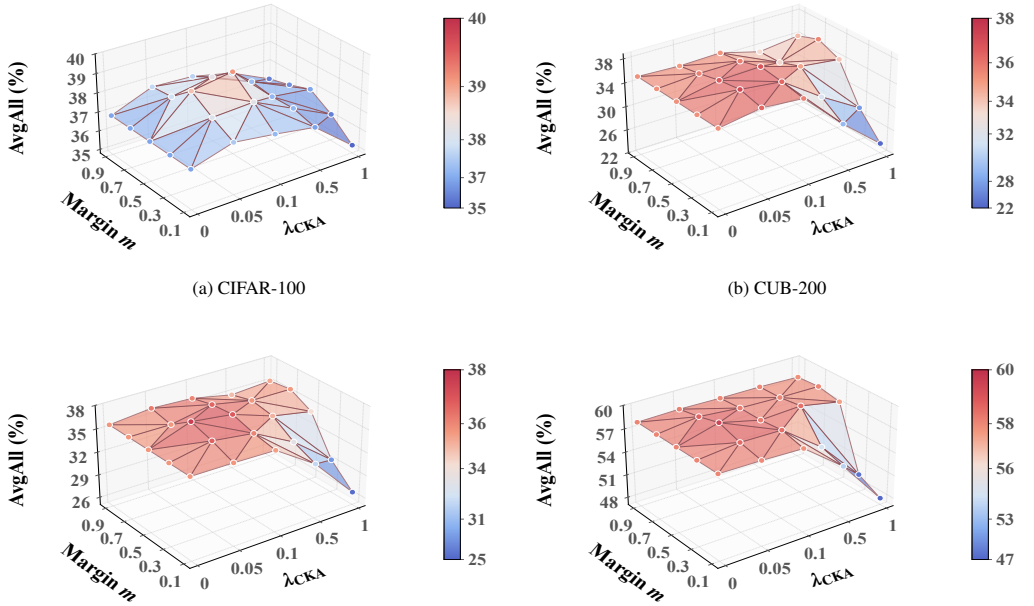
G.1 Joint Effects of the CKA Weight and Margin

Fig. 7 summarizes the joint effects of λ_{CKA} and m on Avg-All across all four benchmarks, evaluating all combinations of $\lambda_{\mathrm{CKA}} \in \{0, 0.05, 0.1, 0.5, 1.0\}$ and $m \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with all other training settings fixed. The best grid configurations are $(0.1, 0.5)$ for CIFAR-100, $(0.05, 0.3)$ for CUB-200, and $(0.05, 0.5)$ for both ImageNet-R and mini-ImageNet. When $\lambda_{\mathrm{CKA}} = 0$, CKA



(a) CIFAR-100 (b) mini-ImageNet

Figure 6: **Attack-wise ablation of FSCIL-AT components.** Clean and attack-specific Top-1 accuracy is reported under different component configurations on CIFAR-100 and mini-ImageNet.



(a) CIFAR-100 (b) CUB-200

(c) ImageNet-R (d) mini-ImageNet

Figure 7: **Sensitivity to the CKA weight λ_{CKA} and margin m.** Avg-All (%) is reported across four datasets under different hyperparameter combinations.

is disabled and m is irrelevant, the No-CKA configuration consistently yields a lower Avg-All than the best-performing CKA configuration across all four datasets. Overall, the highest-performing region is concentrated around small-to-moderate CKA weights and intermediate margins. Small margins under strong regularization can underperform No-CKA, favoring selective suppression over maximal decorrelation, while increasing the margin alleviates this degradation until the constraint becomes too permissive. m determines the tolerance of the hinge penalty on the CKA similarity between the base and incremental residual representations, whereas λ_{CKA} controls its strength in the defender objective. Since the CKA term is evaluated on concatenated CLS and mean-patch residuals at the LoRA injection layers, an overly strict combination can suppress both shared semantic structure and local visual responses that remain useful for adversarial adaptation. This effect is most pronounced on CUB-200, which contains 200 bird species with subtle visual differences, the sharp decline under large λ_{CKA} and small m suggests that excessive decorrelation can suppress the discriminative features needed to distinguish visually similar species.

G.2 Sensitivity to CKA Layer Selection

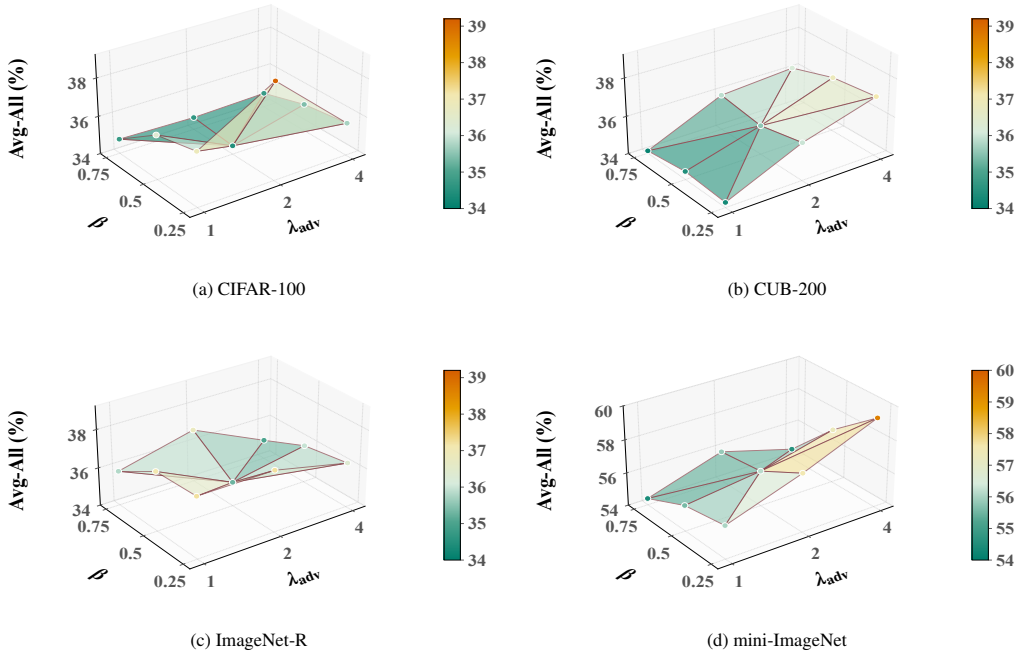
To investigate the impact of CKA layer selection on model performance, we vary $M_{\mathrm{CKA}} \in \{1, 3, 6, 12\}$, where CKA is applied to the last M_{CKA} Transformer blocks while all other set-

tings remain unchanged. Table 7 shows that $M_{\text{CKA}} = 6$ achieves the highest Avg-All on all four datasets. When $M_{\text{CKA}} = 1$ or 3, limited coverage leaves redundancy in the unregularized blocks. At $M_{\text{CKA}} = 12$, CKA also penalizes shared representations in earlier blocks, which encode general visual features, suppressing transferable features needed for adversarial adaptation and reducing performance. The results indicate that applying CKA to an intermediate number of final blocks provides the best balance between suppressing redundant features and preserving transferable visual representations.

Table 7: **Sensitivity to CKA layer selection.** Avg-All (%) is reported when CKA regularization is applied to the last M_{CKA} Transformer blocks.

CKA layers	CIFAR-100	mini-ImageNet	CUB-200	ImageNet-R
1	37.75	56.88	36.21	36.11
3	<u>38.50</u>	57.65	<u>36.26</u>	<u>36.35</u>
6	**39.12**	**59.46**	**37.19**	**37.59**
12	37.16	<u>59.07</u>	36.03	35.74

G.3 Sensitivity to Adversarial Loss Weighting and Allocation



(a) CIFAR-100 (b) CUB-200

(c) ImageNet-R (d) mini-ImageNet

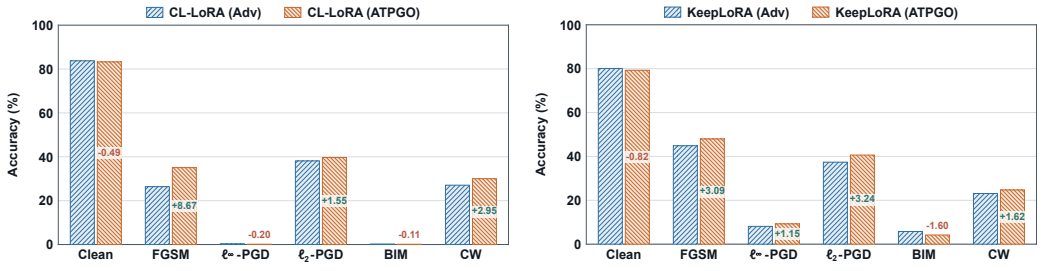
Figure 8: **Sensitivity to λ_{adv} and β across four datasets.** Avg-All (%) is reported for each dataset under different hyperparameter combinations.

Fig. 8 studies the sensitivity of Avg-All to adversarial loss weighting and allocation across four datasets. We vary the total adversarial loss weight $\lambda_{\text{adv}} = \lambda_1 + \lambda_2 \in \{1, 2, 4\}$ and the allocation ratio $\beta = \lambda_1/\lambda_{\text{adv}} \in \{0.25, 0.5, 0.75\}$, where λ_1 and λ_2 weight the transferred and self-generated adversarial losses, respectively. Increasing λ_{adv} does not produce monotonic gains, since larger weights make performance more sensitive to the allocation ratio β. Across the four datasets, the strongest configurations consistently assign a larger share to self-generated supervision. Self generated attacks probe the defender's current decision boundary and provide direct adaptation signals, whereas transferred attacks expose complementary vulnerabilities. A limited transfer contribution is useful, but adominant share weaken salignment with the defender's evolving boundary. The best configurations are $(\lambda_{\text{adv}}, \beta) = (2, 0.25)$ for CIFAR-100 and ImageNet-R and $(\lambda_{\text{adv}}, \beta) = (4, 0.25)$ for mini-ImageNet and CUB-200.

H Generalization of ATPGO to other architectures

To assess the architectural generalization of ATPGO, we incorporate it into two of the most competitive baselines, CL-LoRA and KeepLoRA, and compare their performance with that of the corre-

sponding Naive AT variants. As shown in Fig. 9, ATPGO causes a slight clean accuracy decrease in both methods while improving their overall robust performance. The strongest gains under FGSM, ℓ_2-PGD and CW arise because transferred examples broaden the perturbation patterns presented during training, while self-generated examples adapt supervision to the current expert's local weaknesses, improving coverage beyond the vulnerabilities exposed by standard self-adversarial training. The different ℓ_∞-PGD responses of CL-LoRA and KeepLoRA suggest that architectural differences affect how transferred adversarial supervision is absorbed, thereby influencing both its effectiveness against training-matched ℓ_∞ perturbations and the extent to which it expands adversarial diversity. Despite attack-specific variation, robust-performance gains on both methods show that APTGO transfers to other architectures without relying on FSCIL-AT's specific components.



(a) CL-LoRA

(b) KeepLoRA

Figure 9: **Comparison of Naive AT and ATPGO across architectures on CIFAR-100.** Clean and attack-specific Top-1 accuracy is reported for CL-LoRA and KeepLoRA.

I Evaluation under original backbones and update protocols

To examine whether the comparison depends on the common frozen ViT-B/16 setting, Table 8 reevaluates the baselines with their original backbones and parameter update protocols. Methods whose native configurations already use a frozen ViT remain broadly consistent with the aligned result. In contrast, restoring CNN backbones generally reduces clean and adversarial performance because their different feature geometry and transfer characteristics limit few-shot discrimination and robustness to adversarial shifts. Methods that primarily update classifiers or prototypes also have limited ability to reshape the feature space under this evaluation. FSCIL-AT nevertheless remains the strongest method overall. Its advantage reflects the stronger frozen ViT representation and targeted residual adaptation, in which ATPGO responds to evolving attacks while CKA prevents the adapted residuals from simply copying the base expert.

Table 8: **Clean and adversarial accuracy on CIFAR-100 under the original backbone and parameter update protocol of each method.** Top-1 accuracy (%) is reported. (.) denotes the original backbone.

Method	Train	Clean	Robust accuracy					Avg-All
			FGSM	ℓ_∞-PGD	ℓ_2-PGD	BIM	CW	
LwF	Std	51.72±00.49	04.22±00.13	00.68±00.04	02.51±00.05	02.23±00.05	00.00±00.00	10.23
(AlexNet)	Adv	50.79±00.25	06.03±00.08	00.22±00.06	01.66±00.11	00.78±00.07	00.00±00.00	09.91
RanPAC	Std	72.43±00.03	22.39±00.12	00.02±00.02	24.82±00.27	00.01±00.01	06.36±00.24	21.01
(ViT-B/16)	Adv	72.42±00.07	31.86±00.55	00.81±00.19	27.16±00.14	00.75±00.08	09.43±00.17	23.74
CL-LoRA	Std	**84.86±00.19**	22.63±01.28	00.06±00.03	31.33±00.43	00.08±00.01	12.60±00.45	25.26
(ViT-B/16)	Adv	83.75±00.27	26.39±03.41	00.34±00.09	38.11±01.12	00.23±00.07	27.05±01.22	29.31
KeepLoRA	Std	80.62±00.27	35.27±00.24	03.38±00.28	36.36±00.10	04.04±00.48	12.53±00.15	28.70
(CLIP ViT-B/16)	Adv	80.09±00.31	44.87±00.98	08.11±02.03	37.38±00.55	05.78±01.41	23.09±00.45	33.22
TEEN	Std	05.75±00.80	01.64±00.26	01.53±00.33	04.24±00.49	01.51±00.30	02.31±00.38	02.83
(ResNet-20)	Adv	05.75±00.80	01.64±00.26	01.53±00.28	04.24±00.49	01.51±00.30	02.31±00.38	02.83
BiMC	Std	72.97±00.00	15.71±00.01	00.00±00.00	04.03±00.03	00.00±00.00	03.67±00.03	16.06
(CLIP ViT-B/16)	Adv	73.06±00.04	16.64±00.08	00.00±00.00	04.57±00.08	00.00±00.00	04.20±00.06	16.41
ConCM	Std	40.11±01.20	10.04±00.21	00.19±00.01	00.00±00.00	00.00±00.00	00.00±00.00	08.39
(ResNet-12)	Adv	39.32±01.81	06.82±00.40	00.00±00.00	00.27±00.01	00.00±00.00	00.01±00.01	07.74
AIR (ResNet-34)	Native	02.50±00.10	01.18±00.10	01.04±00.11	01.26±00.08	01.03±00.10	01.24±00.10	01.38
DSEF (ViT-B/16)	Native	01.88±00.49	01.14±00.16	00.42±00.29	00.91±00.53	00.55±00.39	00.49±00.42	00.90
ARMCL (ResNet-18)	Native	48.64±00.45	12.02±00.45	00.28±00.10	01.12±00.16	01.10±00.14	00.00±00.00	10.53
FSCIL-AT (Ours)	**ATPGO**	81.83±00.28	**53.87±00.84**	**11.02±00.97**	**41.68±01.06**	**09.90±00.72**	**36.39±02.07**	**39.12**