RELIABLE ADVERSARIAL CONTINUAL LEARNING VIA ROBUST EXPERT COLLABORATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Adversarially robust continual learning must acquire new tasks while preserving previously learned knowledge and adversarial robustness. These objectives can conflict because adversarial updates to a shared representation may amplify forgetting, induce representation drift, and yield class-imbalanced robust solutions. We propose Robust Expert Collaboration (REC), a replay-free framework that separates task-level robust expert construction from guarded backbone integration. REC first applies Diversified Multi-Expert Learning (DMEL), which jointly optimizes temporary task experts with feature-diversity and gradient-diversity regularization, together with image- and feature-space adversarial objectives, before fusing them into a single task expert. Robust Anchoring and Stability Cooperation (RASC) then adapts the shared backbone using clean and adversarial supervision, clean–adversarial consistency, a frozen teacher feature anchor, and historical decision distillation. A Class-wise Robust Constraint (CRC) selects checkpoints on a held-out gating split using clean accuracy, adversarial accuracy, per-class adversarial recall, teacher feature similarity, and historical decision agreement; when no candidate satisfies the predefined criteria, REC restores the pre-integration checkpoint. Experimental results demonstrate that REC achieves a favorable robustness–retention trade-off while maintaining competitive clean accuracy.

1 INTRODUCTION

Deep neural networks have achieved remarkable success across a broad range of visual recognition tasks. Their deployment in long-lived systems, however, increasingly requires two capabilities at once: continual learning (CL) De Lange et al. (2021); Parisi et al. (2019) and adversarial robustness Goldblum et al. (2020). CL enables a model to acquire knowledge from sequential tasks while limiting catastrophic forgetting Parisi et al. (2019). In parallel, adversarial robustness addresses the susceptibility of neural networks to small, deliberately optimized input perturbations Goodfellow et al. (2014). Combining the two remains difficult because updates that improve robustness on the current task may alter a shared representation used by previously learned tasks.

Although continual learning and adversarial learning have been extensively studied, their combination remains challenging. Existing continual-learning methods, including replay Lopez-Paz & Ranzato (2017); Buzzega et al. (2020); Rebuffi et al. (2017), regularization Kirkpatrick et al. (2017); Zenke et al. (2017), and architectural adaptation, primarily focus on knowledge retention under benign data distributions and do not explicitly preserve adversarial robustness. Meanwhile, adversarial training improves robustness in stationary settings by optimizing models on adversarially perturbed inputs Goodfellow et al. (2014); Jia et al. (2022); Madry et al. (2017), but repeatedly applying such updates during continual adaptation can interfere with representations shared across tasks.

The difficulty stems from three tightly coupled challenges. First, continual learning requires simultaneously maintaining plasticity for learning new tasks and stability for preserving previous knowledge Dohare et al. (2024), whereas adversarial training can introduce updates that intensify forgetting. Second, robust optimization exhibits a trade-off between clean accuracy and adversarial robustness Deng & Mu (2024). This trade-off becomes more difficult when sequential tasks share a feature backbone and updates for a new task shift representations used by historical experts. Third, adversarial continual-learning methods often optimize one evolving model throughout the task sequence Zhou & Hua (2024); Mi et al. (2025). Such optimization can produce correlated adversarial vulnerabilities, degrade previously learned decision boundaries, and, in extreme cases, map adversarial inputs from

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.

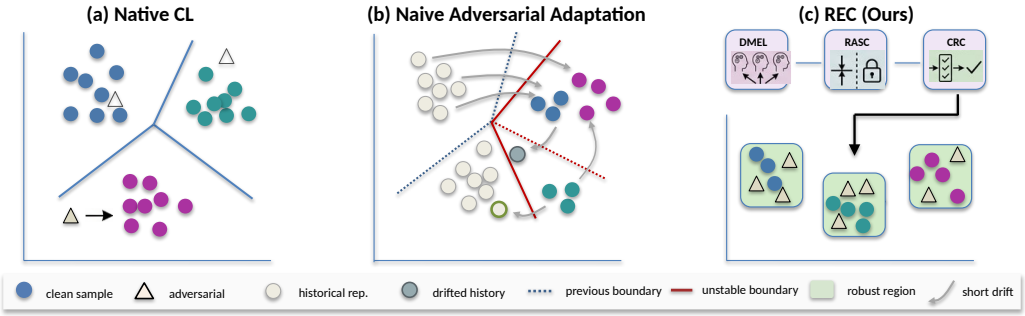


Figure 1: Motivation for separating robust expert learning from backbone integration. **(a)** Clean separation can remain adversarially fragile. **(b)** Naive adversarial adaptation drifts historical representations. **(c)** REC diversifies, anchors, and validates robust integration.

multiple classes to one dominant category. These observations motivate separating task-level robust expert construction from shared-backbone integration.

Figure 1 summarizes this difficulty. A native continual learner may retain well-separated clean representations, yet small adversarial perturbations can still cross its task-local decision boundaries. Directly applying adversarial optimization to the shared backbone does not fully resolve the problem: updates driven by the current task can shift representations used by historical experts, distort old–new task boundaries, and occasionally produce checkpoints whose adversarial predictions concentrate on a small subset of classes. Robust continual learning therefore requires more than appending an adversarial-training stage to a clean continual learner.

To address these challenges, as illustrated in Figure 1(c), we propose Robust Expert Collaboration (REC), a replay-free framework for adversarially robust task-incremental learning. Rather than relying on one unconstrained adversarial update, REC separates expert diversification, shared-backbone integration, and checkpoint validation. This decomposition introduces task plasticity locally while guarding the shared representation and the class balance of the retained robust solution.

We first propose Diversified Multi-Expert Learning (DMEL) to construct multiple collaborative experts for each incoming task. Starting from a shared clean initialization, the experts are jointly optimized instead of being trained independently. An HSIC penalty discourages dependent intermediate representations, while GradDiv penalizes alignment between loss gradients with respect to the frozen backbone feature. Dual-level adversarial training then optimizes each temporary expert against both pixel-space PGD and a local feature-space PGD neighborhood. Finally, uniform parameter averaging merges the temporary branches into one task expert, so collaboration does not increase the number of experts used at inference for a given task.

To limit forgetting during backbone integration, we further propose Robust Anchoring and Stability Cooperation (RASC), which regularizes robust backbone adaptation using a frozen teacher snapshot. RASC jointly employs feature cosine anchoring, historical decision distillation on clean and adversarial views, and prediction- and feature-consistency terms between clean and adversarial inputs. These constraints are designed to reduce representation and decision drift while the shared backbone and current expert are updated.

Finally, we introduce a Class-wise Robust Constraint (CRC) for validation-based checkpoint selection. CRC evaluates clean accuracy, adversarial accuracy, class-wise recall, teacher similarity, and historical decision agreement on a held-out gating split. If no candidate satisfies the criteria, REC restores the pre-integration checkpoint, reducing the risk of selecting class-collapsed solutions without claiming guarantees on unseen test data.

Experiments on Split CIFAR-10 and Split CIFAR-100 evaluate REC under a unified Task-IL protocol, with additional strong-attack and continual-learning diagnostics on Split CIFAR-10. The primary contributions are: (1) We propose Robust Expert Collaboration (REC), a replay-free framework that combines HSIC-based feature diversity, adversarial gradient diversity, and dual-level adversarial training to construct and fuse robust task experts; (2) We propose Robust Anchoring and Stability Cooperation (RASC), which combines frozen-teacher feature anchoring, historical decision preservation, and clean–adversarial consistency to constrain shared-backbone adaptation; (3) We develop a Class-wise Robust Constraint (CRC), a validation-based safety mechanism that selects checkpoints

using clean accuracy, PGD accuracy, per-class adversarial recall, teacher similarity, and historical decision agreement, with rollback when no candidate satisfies the gating criteria.

2 Related Work

Continual Learning (CL) aims to acquire sequential tasks without catastrophic forgetting. Existing methods include replay, regularization, architecture-based isolation, and pretrained-model adaptation. Replay methods retain a bounded memory of past data Lopez-Paz & Ranzato (2017); Chaudhry et al. (2018); Buzzega et al. (2020); Rebuffi et al. (2017), while regularization and architectural methods mitigate interference through constrained updates or task-specific components. Recent Vision Transformer-based approaches achieve parameter-efficient continual learning via prompts, prototypes, adapters, and low-rank adaptation Wang et al. (2022b); Smith et al. (2023); Wang et al. (2022a); McDonnell et al. (2024); He et al. (2025); Li et al. (2025); Chen et al. (2025); Wang et al. (2025); Chen et al. (2026); Luo et al. (2026); Lou et al. (2026). However, they primarily focus on clean-data retention and parameter efficiency, leaving the robustness of task-wise knowledge under adversarial perturbations largely unexplored.

Adversarial Continual Learning. Adversarial continual learning jointly addresses knowledge retention and adversarial robustness through replay, pseudo replay, robust knowledge transfer, or robustness-aware optimization. Defense without Forgetting uses anisotropic and isotropic pseudo replay Zhou & Hua (2024), while the Adversarial Robust Memory-Based Continual Learner evaluates both clean and adversarial forgetting under bounded replay Mi et al. (2025). FLAIR analyzes the conflict between adversarial training and incremental adaptation and promotes flatter robust solutions Cho et al. (2025).

A related recent approach, USP, addresses semi-supervised continual learning through a divide-and-conquer framework and Feature Space Reservation (FSR) Duan et al. (2025). FSR reserves feature-space locations for future classes to improve learning plasticity. Since the original USP is designed for semi-supervised continual learning rather than adversarial robustness, we construct USP-FSR-AT by adapting its FSR mechanism to our fully supervised Task-IL setting under the unified adversarial-training protocol.

These studies highlight the importance of evaluating robustness throughout the task sequence. Moreover, aggregate adversarial accuracy can obscure severe class-wise failures Zhang et al. (2024). REC differs from these approaches by retaining no historical training examples, constructing diversified task-local robust experts with DMEL, constraining shared-backbone integration with RASC, and applying CRC for class-sensitive checkpoint validation and rollback.

3 Methodology

3.1 Problem Definition

In CL, a model is trained on a sequence of tasks in an incremental manner, where knowledge from earlier tasks is typically not fully accessible during subsequent training stages. Formally, we consider a task stream $\mathcal{T} = \{1, \ldots, n'\}$, where each task t is associated with a training set $D_t = (\mathbf{x}_c, \mathbf{y}_c)_{c=1}^{n_t}$ and n' denotes the total number of tasks. Here, $\mathbf{x}_c \in \mathcal{X}$ denotes an input image and $\mathbf{y}_c \in \mathcal{Y}$ represents its corresponding class label, with n_t denoting the total number of samples in D_t. In standard CL settings, each task provides clean samples, and the central challenge is to mitigate catastrophic forgetting while preserving performance on previously learned tasks. During evaluation, the trained model is tested against multiple white-box attacks using all testing samples.

3.2 Diversified Multi-Expert Learning

Achieving adversarial robustness in continual learning is substantially harder than in the static setting for two reasons. First, the backbone must balance *plasticity* for new tasks and *stability* against forgetting, a tension exacerbated by adversarial training with large, prolonged gradient updates. Second, the inherent clean–robustness trade-off Zhang et al. (2019) becomes more challenging in sequential learning, where aggressive adversarial updates can distort the shared representation and permanently degrade previously learned class boundaries.

A natural response is to seek a *diverse ensemble* of models that collectively span a wide robustness manifold, then distill their complementary strengths into a single compact expert. However, naïvely training independent models from random initializations in a continual setting introduces substantial variance and wastes computation. More critically, independently trained models tend to converge to *correlated* adversarial blind spots because they are all trained on the same data distribution with the same loss and their input-space attack gradients align closely, meaning a single transferable

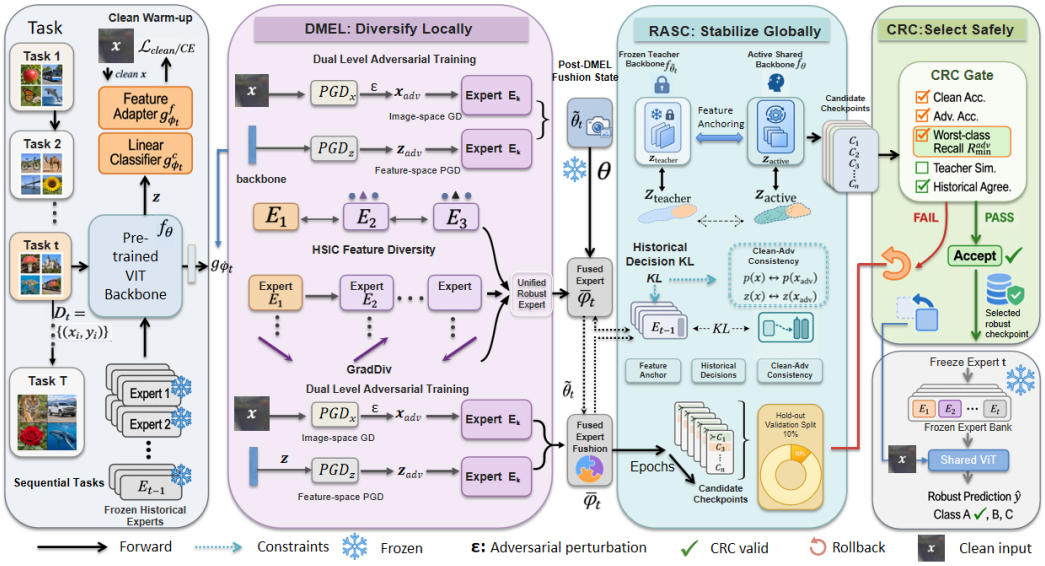


Figure 2: REC overview. At task t, a clean-warmed expert is replicated and diversified by DMEL with image- and feature-space adversarial training, then uniformly fused. RASC integrates the fused expert into the shared representation under a frozen teacher and historical decision constraints. CRC selects a class-feasible checkpoint on a held-out gate; otherwise, it restores the post-DMEL fusion. The accepted expert is frozen in the historical expert bank for later tasks.

perturbation can fool the entire ensemble simultaneously. In this paper,we propose a novel Diversified Multi-Expert Learning (DMEL) mechanism to optimize various experts to enhance the model's generalization performance.

Specifically, we adopt a pretrained Vision Transformer (ViT) as the shared backbone f_θ, which maps an input image $\mathbf{x}$ to a feature representation $\mathbf{z} = f_\theta(\mathbf{x})$. For each incremental task t, we dynamically create a dedicated task expert g_{ϕ_t}, comprising a feature adaptation layer $g_{\phi_t^f} : \mathcal{X} \to \mathcal{Z}'$ and a task-specific linear classifier $g_{\phi_t^c} : \mathcal{Z}' \to \mathcal{Y}$, where $\mathcal{Z}'$ and $\mathcal{Y}$ denote the feature and prediction space, respectively. The feature vector extracted from the shared backbone f_θ is fed into $g_{\phi_t^f}$ to produce a feature vector $\mathbf{z}' \in \mathcal{Z}'$ that is used as an input of $g_{\phi_t^c}$ to produce a final prediction $\mathbf{y} \in \mathcal{Y}$.

Clean Pre-training and Expert Replication. Upon the arrival of task t, we first create the new expert g_{ϕ_t} and treat it as a base expert, which is trained on the clean data to establish a stable initialization:

$$\mathcal{L}_{\text{CE}}(g_{\phi_t}, D_t) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}\mathbf{y}_i[c]\log\left[g_{\phi_t^c}\left(g_{\phi_t^f}\left(f_\theta(\mathbf{x}_i)\right)\right)\right]_c, \tag{1}$$

where $\mathbf{y}_i[c]$ denotes the c-th element of the one-hot label $\mathbf{y}_i$, and $\left[g_{\phi_t^c}(g_{\phi_t^f}(f_\theta(\mathbf{x}_i)))\right]_c$ denotes the predicted probability for class c. Directly applying strong adversarial training from a random initialization can lead to unstable optimization and class imbalance. We therefore first perform clean warm-up to obtain a stable initialization. Once the base expert converges, it is replicated into K collaborative expert branches $\{g_{\phi_t(1)}, \ldots, g_{\phi_t(K)}\}$, which are then jointly optimized under diversity and robustness constraints.

HSIC-Based Feature Diversity. Since all experts share the same initialization, without an explicit diversity objective they tend to converge to nearly identical solutions. To address this, we promote statistical independence between the intermediate representations of different experts. Let $\mathbf{h}_k^t = g_{\phi_t^f(k)}(f_\theta(\mathbf{x}))$ denote the intermediate feature of the k-th collaboration expert. We employ the Hilbert-Schmidt Independence Criterion to penalize pairwise statistical dependence:

$$\mathcal{L}_{\text{HSIC}} = \frac{1}{K(K-1)}\sum\nolimits_{i\neq j}\text{HSIC}(\mathbf{h}_i^t, \mathbf{h}_j^t), \tag{2}$$

where $\text{HSIC}(\cdot,\cdot)$ is the HSIC measure function. Minimizing $\mathcal{L}_{\text{HSIC}}$ encourages complementary feature representations among experts and may reduce vulnerability correlation under adversarial perturbations, providing richer information for subsequent parameter fusion.

Adversarial Gradient Diversity (GradDiv). Feature diversity alone is insufficient in which two experts with distinct intermediate representations may still produce highly correlated input-space attack gradients, making the ensemble vulnerable to a single transferable adversarial direction. Let $\mathbf{g}_k^t = \nabla_{\mathbf{x}} \mathcal{L}_{\mathrm{CE}}(g_{\phi_t(k)}, \mathbf{y})$ be the loss gradient of the k-th collaboration expert with respect to the input during the t-th task learning. We introduce a GradDiv regularizer that penalizes cosine alignment between expert gradients:

$$\mathcal{L}_{\text{GradDiv}} = \frac{1}{K(K-1)} \sum\nolimits_{i \neq j} \left\{ \frac{\sum_{c=1}^{n} \mathbf{g}_i^t[c] \mathbf{g}_j^t[c]}{\sqrt{\sum_{c=1}^{n} (\mathbf{g}_i^t[c])^2} \sqrt{\sum_{c=1}^{n} (\mathbf{g}_j^t[c])^2}} \right\}, \tag{3}$$

where $\mathbf{g}_i^t[c]$ and $\mathbf{g}_j^t[c]$ denote the c-th dimension of $\mathbf{g}_i^t$ and $\mathbf{g}_j^t$, respectively. n is the total number of dimensions within $\mathbf{g}_i^t$. By minimizing gradient cosine similarity, adversarial perturbations crafted to fool one expert are unlikely to transfer effectively to the others, substantially improving ensemble robustness.

Dual-Level Adversarial Training. Conventional adversarial training perturbs inputs only in pixel space. We argue this overfits to a single attack manifold and fails to harden the learned representations against semantic-level perturbations. We therefore introduce a *dual-level* adversarial training scheme that simultaneously generates perturbations in (i) the input image space and (ii) the backbone feature space.

Let us define $\delta^{s+1}(k)$ as a perturbation in the image space, produced by the k-th collaborative expert at the $(s+1)$-th training epoch. We employ the PGD as the adversarial sample generator and the adversarial perturbations are produced under a ℓ_∞ constraint:

$$\delta^{s+1}(k) = \Pi_{[-\epsilon,\epsilon]} \left[\delta^s(k) + \alpha \operatorname{sign}(\nabla_{\mathbf{x}} \mathcal{L}_{\mathrm{CE}}(g_{\phi_t(k)}, D_t)) \right], \quad \tilde{\mathbf{x}}_k^t = \operatorname{clip}(\mathbf{x} + \delta(t),\, 0, 1), \tag{4}$$

where the operator $\Pi_{[-\epsilon,\epsilon]}(\cdot)$ represents the projection onto the ℓ_∞ ball with radius ϵ, ensuring that the perturbation satisfies $\|\delta\|_\infty \leq \epsilon$. Here, ϵ denotes the maximum perturbation budget and is set to $8/255$, whereas $\alpha = 2/255$ is the PGD step size. The function $\operatorname{sign}(\cdot)$ returns the element-wise sign of its input. The term $\nabla_{\mathbf{x}} \mathcal{L}_{\mathrm{CE}}(g_{\phi_t}, D_t)$ denotes the gradient of the cross-entropy loss with respect to the input image $\mathbf{x}$, where $\mathcal{L}_{\mathrm{CE}}$ is the cross-entropy loss. The adversarial example is generated as $\tilde{\mathbf{x}}_k^t = \operatorname{clip}(\mathbf{x} + \delta(k), 0, 1)$, where $\mathbf{x}$ is the original image, $\delta(k)$ is the generated adversarial perturbation, and $\operatorname{clip}(\cdot, 0, 1)$ constrains the pixel values of the adversarial image to the valid range $[0, 1]$.

In addition to the perturbations in the image space, we also enforce sample diversity by producing perturbations on the feature space from the static frozen backbone.

$$\eta^{s+1}(k) = \Pi_{[-\epsilon,\epsilon]} [\eta^s(k) + \alpha \operatorname{sign}(\nabla_{\mathbf{z}} \mathcal{L}_{\mathrm{CE}}(g_{\phi_t}, D_t))], \quad \tilde{\mathbf{z}}_k^t = \mathbf{z} + \delta(k), \tag{5}$$

where $\eta^{s+1}(k)$ is a generated perturbation in the shared feature space updated at the $(s+1)$-th training epoch and $\mathbf{z}_k^t$ is the adversarial example. $\mathbf{z}$ is the feature vector extracted from the backbone. For a given input $\mathbf{x}$, we employ Eq. (4) and Eq. (5) to produce a paired adversarial sample $\{\tilde{\mathbf{x}}_k^t, \tilde{\mathbf{z}}_k^t\}$ using the k-th collaboration expert during the t-th task learning. Based on generated adversarial samples, we design an adversarial training loss function, expressed as:

$$\mathcal{L}_{\mathrm{adv}} = \lambda_{\text{img}} \frac{1}{K} \sum\nolimits_{k=1}^{K} \left\{ \mathcal{L}_{\mathrm{CE}}(g_{\phi_t(k)}, \tilde{\mathbf{x}}_k^t, \mathbf{y}) \right\} + \lambda_{\text{feat}} \frac{1}{K} \sum\nolimits_{k=1}^{K} \left\{ \tilde{\mathcal{L}}_{\mathrm{CE}}(g_{\phi_t(k)}, \tilde{\mathbf{z}}_k^t, \mathbf{y}) \right\}, \tag{6}$$

where $\mathcal{L}_{\mathrm{CE}}(g_{\phi_t(k)}, \tilde{\mathbf{x}}_k^t, \mathbf{y})$ denotes the cross-entropy loss calculated on the adversarial sample $\tilde{\mathbf{x}}_k^t$ and $\tilde{\mathcal{L}}_{\mathrm{CE}}(g_{\phi_t(k)}, \tilde{\mathbf{z}}_k^t, \mathbf{y})$ is defined as $\sum_{c=1}^{C} \{\mathbf{y}[c] \log g_{\phi_t^c(k)}(g_{\phi_t^f(k)}(\tilde{\mathbf{z}}_k^t))[c]$. λ_{img} and λ_{feat} denote the hyperparameters used to control the importance of the image- and feature-level perturbations. While image-space adversarial training builds pixel-level robustness, feature-space adversarial training compels the expert heads to remain stable under semantic-level representation shifts. Together, they ensure robustness across a substantially larger perturbation manifold than either alone. The final cooperative training objective function for all collaboration experts and the current base expert is defined as:

$$\mathcal{L}_{\text{DMEL}} = \mathcal{L}_{\text{clean}} + \lambda_{\text{hsic}}\, \mathcal{L}_{\text{HSIC}} + \lambda_{\text{grad}}\, \mathcal{L}_{\text{GradDiv}} + \mathcal{L}_{\mathrm{adv}}\,, \tag{7}$$

where λ_{hsic} and λ_{grad} denote the hyperparameters used to control the importance of the feature and gradient diversity terms, respectively. $\mathcal{L}_{\mathrm{clean}}$ is the loss calculated by all K cooperative experts:

$$\mathcal{L}_{\text{clean}} = \frac{1}{K} \sum\nolimits_{k=1}^{K} \mathcal{L}_{\mathrm{CE}}(g_{\phi_t(k)}, D_t)\,. \tag{8}$$

This term ensures that diversity-inducing constraints do not degrade clean classification ability. Upon convergence of the K cooperative experts, we fuse their parameters into a single task expert via uniform averaging:

$$\bar{\phi}_t = \frac{1}{K} \sum\nolimits_{k=1}^{K} \phi_t^k. \tag{9}$$

Uniform fusion is preferred over learned weighted fusion because the individual expert weights tend to be nearly equal in practice (indicating balanced contributions), and weighted variants risk over-weighting experts with high clean accuracy but poor robustness, thereby undermining the carefully cultivated diversity.

3.3 Robust Anchoring and Stability Cooperation

Although DMEL provides each task with a diverse, robustly initialized expert, integrating the fused parameters $\bar{\phi}_t$ into the shared backbone via adversarial fine-tuning introduces several stability challenges in continual learning.

First, large adversarial updates can distort the shared representation, potentially degrading previously learned experts when historical training data are unavailable for correction.

Second, unconstrained adversarial fine-tuning may drive the model toward a degenerate solution in which adversarial predictions concentrate on a small subset of classes. We refer to this failure mode as robust collapse. This failure mode can be obscured by aggregate adversarial accuracy, since high overall accuracy does not necessarily imply balanced class-wise robustness. Once class boundaries are severely distorted by backbone drift, recovering the affected expert can require substantial additional retraining.

Third, robust collapse stems from feature, decision, and prediction drift, affecting representation geometry, historical expert outputs, and clean–adversarial consistency, respectively. To address these complementary failures, we propose Robust Anchoring and Stability Cooperation (RASC), which employs a frozen teacher, a feature-space cosine anchor, and a historical-decision KL anchor to stabilize all three levels.

After DMEL completes and the K experts are fused via Eq. (9), we record a frozen copy of the backbone as the teacher $\tilde{\theta}_t \leftarrow \theta$. The teacher module $g_{\tilde{\theta}_t}$ does not participate in gradient updates during subsequent robust integration and provides a stable reference point that grounds the current active backbone g_θ throughout adversarial fine-tuning.

Feature Cosine Anchoring. For a given data sample $\mathbf{x}$, let us define $\tilde{\mathbf{z}}_{\text{share}}$ and $\mathbf{z}_{\text{share}}$ as the feature representations extracted by $f_{\tilde{\theta}_t}$ and f_θ, respectively. We constrain the current backbone's representations on both clean and adversarial inputs by:

$$\mathcal{L}_{\text{teacher}} = \frac{1}{2}\Big[1 - \cos\big(f_{\tilde{\theta}_t}(\mathbf{x}),\ f_\theta(\mathbf{x})\big) + 1 - \cos\big(f_{\tilde{\theta}_t}(\tilde{\mathbf{x}}),\ f_\theta(\tilde{\mathbf{x}})\big)\Big], \tag{10}$$

where $\tilde{\mathbf{x}}$ is the adversarial sample generated using the current active expert $g_{\bar{\phi}_t}$. This soft constraint permits the backbone to adapt toward robustness while bounding the magnitude of representation drift, preserving both clean discriminability and previously acquired task knowledge.

Historical Decision KL Anchoring. Once the DMEL completes, the proposed framework maintains frozen historical experts whose decision boundaries must be preserved as the shared backbone continues to be refined. To reduce negative knowledge transfer effects, we distill the soft output distributions of the frozen backbone into the current model using temperature-scaled KL divergence, applied to both clean and adversarial inputs:

$$\begin{aligned}\mathcal{L}_{\text{old}} = \frac{T^2}{t-1} \sum\nolimits_{s=1}^{t-1} \big\{ & D_{KL}(P(g_{\bar{\phi}_t^f}(f_{\tilde{\theta}_s}(\mathbf{x}))\,|\,\mathbf{y}) \,||\, P(g_{\bar{\phi}_s^f}(f_\theta(\mathbf{x}))\,|\,\mathbf{y})) \\ & + D_{KL}(P(g_{\bar{\phi}_t^f}(f_{\tilde{\theta}_s}(\tilde{\mathbf{x}}^s))\,|\,\mathbf{y}) \,||\, P(g_{\bar{\phi}_s^f}(f_\theta(\tilde{\mathbf{x}}^s))\,|\,\mathbf{y}))\big\},\end{aligned} \tag{11}$$

where $P(g_{\bar{\phi}_t^f}(f_{\tilde{\theta}_s}(\mathbf{x}))\,|\,\mathbf{y})$ and $P(g_{\bar{\phi}_s^f}(f_\theta(\mathbf{x}))\,|\,\mathbf{y})$ denote the predictive distributions implemented using $g_{\bar{\phi}_t^c}(g_{\bar{\phi}_t^f}(f_{\tilde{\theta}_t}(\mathbf{x})))$ and $g_{\bar{\phi}_t^c}(g_{\bar{\phi}_t^f}(f_\theta(\mathbf{x}))))$, respectively. $\tilde{\mathbf{x}}^s$ is the adversarial sample generated using the s-th expert. T denotes the distillation temperature. The T^2 scaling compensates for the softened gradient magnitudes introduced by temperature scaling. This term directly prevents shared-backbone updates from eroding the adversarial decision boundaries of historical task experts.

Table 1: Final Task-IL accuracy (%) under the unified post-resize 224×224 evaluation. RobustAvg is the arithmetic mean of the four adversarial accuracies. Bold and underlined entries denote the best and second-best results, respectively, within each dataset block. For adapted baselines, the suffix "-AT" denotes evaluation with the unified adversarial-training protocol, while the venue indicates the source continual-learning method.

			Clean	Adversarial accuracy ↑				
Dataset	Method	Replay imgs.	Acc. ↑	FGSM	PGD-ℓ_∞	PGD-ℓ_2	BIM	RobustAvg ↑
Split CIFAR-10	**REC (Ours)**	0	95.05	**78.58**	**72.07**	**94.18**	**72.46**	**79.32**
	USP-FSR-AT (ICCV'25)	0	93.44	<u>77.54</u>	<u>71.82</u>	92.53	<u>71.51</u>	<u>78.35</u>
	FLAIR-AT (AAAI'25)	0	94.22	76.43	70.23	<u>93.17</u>	69.80	77.41
	DSGD-FS-AT (AAAI'24)	500	98.26	51.52	42.29	71.86	39.02	51.17
	CaRE-FS-AT (ICML'26)	0	98.10	58.11	33.71	32.62	32.57	39.25
	CaRE-FS (ICML'26)	0	97.89	65.63	0.01	25.11	0.02	22.69
	DSGD-FS (AAAI'24)	500	<u>99.27</u>	40.88	0.00	46.11	0.00	21.75
	DER-ViT	2000	**99.30**	33.60	0.00	51.29	0.00	21.22
	DRIFT-TIL	0	78.80	19.23	5.47	44.26	4.99	18.49
Split CIFAR-100	**REC (Ours)**	0	92.39	**67.36**	**48.97**	**84.84**	<u>47.78</u>	**61.94**
	USP-FSR-AT (ICCV'25)	0	85.33	<u>54.12</u>	47.18	83.48	46.68	57.87
	FLAIR-AT (AAAI'25)	0	86.68	55.26	<u>48.58</u>	<u>84.79</u>	**48.01**	<u>59.16</u>
	DSGD-FS-AT (AAAI'24)	2000	89.11	48.50	28.53	70.35	33.67	45.26
	CaRE-FS-AT (ICML'26)	0	95.94	29.99	0.14	61.37	0.12	22.91
	CaRE-FS (ICML'26)	0	**97.55**	46.48	0.00	18.16	0.00	16.16
	DSGD-FS (AAAI'24)	2000	92.47	2.16	0.00	16.47	0.00	4.66
	DER-ViT	2000	<u>97.51</u>	24.23	0.00	56.09	0.00	20.08

Internal Robustness Consistency. Beyond cross-model anchoring, we also enforce internal consistency of the current active model, expressed as:

$$\begin{aligned}\mathcal{L}_{\text{TRADES}} = T^2\, D_{KL}(P(g_{\bar{\phi}_t^f}(f_\theta(\mathbf{x}))\,|\,\mathbf{y})\,||\,P(g_{\bar{\phi}_t^f}(f_\theta(\tilde{\mathbf{x}}^t))\,|\,\mathbf{y})) \\ - \cos\Big(g_{\bar{\phi}_t^f}(f_\theta(\mathbf{x})),\, g_{\bar{\phi}_t^f}(f_\theta(\tilde{\mathbf{x}}^t))\Big)\,,\end{aligned} \tag{12}$$

where the first term penalizes prediction inconsistency between clean and adversarial inputs of the same model and the second term encourages the backbone to produce geometrically consistent representations for natural and perturbed versions of the same sample. These two terms operate within the current model and are distinct from the teacher anchoring losses. The complete loss for the RASC robust integration stage is defined as:

$$\mathcal{L}_{\text{RASC}} = \mathcal{L}_{\text{clean}} + \lambda_f\, \mathcal{L}_{\text{teacher}} + \lambda_o\, \mathcal{L}_{\text{old}} + \lambda_{\text{TRADES}} \mathcal{L}_{\text{TRADES}}, \tag{13}$$

where λ_f, λ_o and λ_{TRADES} are hyperparameters used to control the contributions of the last three loss terms.

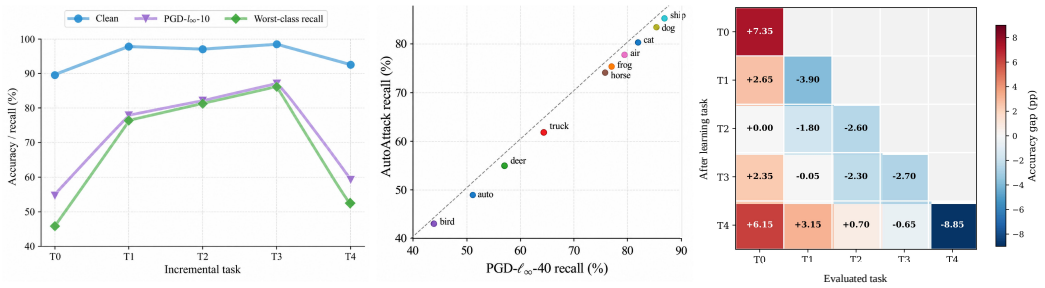
3.4 Class-wise Robust Constraint

Although RASC improves the stability of robust integration, it remains an optimization objective and cannot guarantee that every optimization step benefits the final continual learner. In practice, adversarial optimization may converge to degenerate checkpoints that satisfy the global objective while suffering from severe class imbalance under attack. Since task experts are frozen after training, selecting checkpoints solely based on overall adversarial accuracy can permanently compromise the robustness of affected classes, as a model may achieve competitive aggregate accuracy by performing well on only dominant classes while failing on others.

To eliminate this failure mode, we introduce a **Class-wise Robust Constraint (CRC)**, which acts as a safety mechanism during robust integration. Unlike DMEL and RASC, CRC is not an optimization objective. Instead, it evaluates every candidate checkpoint on a held-out validation set and accepts only those satisfying a collection of robustness constraints. If none of the candidate checkpoints satisfy all constraints, optimization is discarded and the model is restored to the fused checkpoint produced by DMEL.

Per-Class Adversarial Recall. Let R_c^{adv} denote the adversarial recall of class c, which measures the fraction of true class-c samples that are correctly classified under adversarial attack-i.e., the per-class true positive rate in the adversarial setting, defined as:

$$R_c^{\text{adv}} = \frac{\#\{\hat{y}_i = c,\ y_i = c\}}{\#\{y_i = c\}}\,, \tag{14}$$



(a) Task-Level Robustness (b) Strong-Attack Consistency (c) Robust BWT Matrix

Figure 3: Task-, class-, and stream-level robustness. **(a)** Clean accuracy, PGD-ℓ_∞-10 accuracy, and worst-class recall across tasks. **(b)** Per-class recall under five-restart PGD-ℓ_∞-40 and task-local binary AutoAttack. **(c)** Robust backward transfer across stages.

where $\#\{\hat{y}_i = c,\ y_i = c\}$ denotes the cardinality of the set of samples that are both truly from class c and correctly predicted as class c under adversarial attack. $\#\{y_i = c\}$ denotes the cardinality of the set of all samples whose ground-truth label is c. We define the worst-case adversarial recall as $R^{\text{adv}}_{\min} = \min_c R^{\text{adv}}_c$. A model checkpoint is only accepted as a valid candidate if it simultaneously satisfies:

$$\text{Clean} \geq \tau_{\text{clean}}, \quad \text{AdvAcc} \geq \tau_{\text{adv}}, \quad R^{\text{adv}}_{\min} \geq \tau_{\text{recall}}\,, \tag{15}$$

where Clean denotes the classification accuracy on the held-out gating split, and τ_{clean} is the corresponding clean-accuracy threshold. AdvAcc denotes the adversarial accuracy evaluated on the same split, with τ_{adv} denoting its threshold. $R^{\text{adv}}_{\min}$ denotes the minimum per-class adversarial recall on the gating split, and τ_{recall} is the corresponding recall threshold. We construct the gating set $\widetilde{D}_t$ by holding out 10% of the current task training set D_t, and all checkpoint-selection statistics are computed exclusively on $\widetilde{D}_t$. The test set is used only for final evaluation.

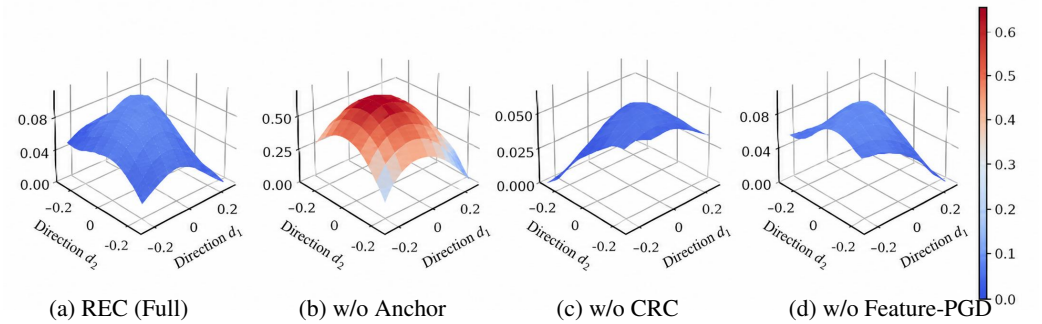
Multi-Criteria Candidate Selection and Rollback. Among all epochs that satisfy the gating conditions in Eq. (15), we select the candidate checkpoint according to its validation performance in terms of adversarial accuracy, clean accuracy, and historical decision consistency. If no epoch satisfies all gating conditions, the model is restored to the pre-robust-integration fusion checkpoint $\phi_t \leftarrow \bar{\phi}_t$. This rollback rule avoids accepting a robust-integration checkpoint that violates the predefined gating criteria and instead falls back to the post-DMEL fusion state. However, it does not provide a formal guarantee on unseen test data. In this way, CRC reduces the risk of retaining severely class-imbalanced or robustness-degraded checkpoints. Together, DMEL, RASC, and CRC form a guarded robust-integration pipeline that combines expert diversity, representation stabilization, and validation-based checkpoint selection.

4 Experiments

We evaluate REC under the fully supervised Task-Incremental Learning (Task-IL) protocol on Split CIFAR-10, comprising five two-class tasks, and Split CIFAR-100, comprising ten ten-class tasks. Unless otherwise specified, all methods are evaluated under a unified protocol using the same CLIP-pretrained ViT-B/16 backbone, 224×224 input resolution, task splits, training budget, and task-aware white-box evaluation. We report clean accuracy and adversarial accuracy under FGSM, PGD-ℓ_∞, PGD-ℓ_2, and BIM. RobustAvg denotes the arithmetic mean of the four adversarial accuracies. Detailed reimplementation status and evaluation settings are provided in Appendix A.1.

4.1 Comparison with Prior Methods

Table 1 compares final Task-IL performance under the unified evaluation protocol. REC achieves the highest RobustAvg on Split CIFAR-10 (79.32%) while retaining 95.05% clean accuracy without replay. It improves PGD-ℓ_∞/RobustAvg over USP-FSR-AT by 0.25/0.97 points and over FLAIR-AT by 1.84/1.91 points, indicating a balanced gain rather than an advantage confined to a single attack. Two regimes emerge from the results. Native continual-learning baselines maintain 97.89–99.30% clean accuracy but perform poorly under iterative ℓ_∞ attacks, indicating limited adversarial retention. Adversarial training substantially improves robustness, yet DSGD-FS-AT and CaRE-FS-AT still trail REC by 29.78 and 38.36 PGD-ℓ_∞ points, respectively, suggesting that simply appending adversarial fine-tuning is insufficient to match REC's robust retention under the unified protocol.



(a) REC (Full) (b) w/o Anchor (c) w/o CRC (d) w/o Feature-PGD

Figure 4: Local adversarial-loss geometry around each checkpoint in the last four ViT blocks. d_1 and d_2 denote normalized filter-wise perturbation directions; all panels share the same loss scale.

4.2 Robustness and Continual Diagnostics

Full strong-attack, continual-retention, and three-seed reproducibility results are reported in Appendix B. Here we retain task- and class-level diagnostics that explain the aggregate behavior.

Figure 3 analyzes robustness at the task, class, and stream levels. While clean accuracy remains consistently high, PGD accuracy is lowest on the first and last tasks, reflecting distinct continual-learning pressures. The worst-class curve indicates residual class imbalance rather than task collapse.

Panel (b) tests whether the observed class pattern is specific to one iterative attack. Five-restart PGD-ℓ_∞-40 and task-local binary AutoAttack show qualitatively similar class-recall patterns, while bird remains difficult under both. Their agreement supports the observed class-level trend, but does not establish security against every adaptive attack. Panel (c) localizes the stream-level difference between REC and USP-FSR-AT through the stage-by-task PGD-ℓ_∞-10 accuracy gap (REC minus USP-FSR-AT). At the final stage, REC improves Tasks 0–2 by $+6.15$, $+3.15$, and $+0.70$ points, respectively, remains close on Task 3 (-0.65), but trails on the newly introduced Task 4 (-8.85).

Table 2: Component ablation under the unified Split CIFAR-10 protocol. A checkmark indicates that the component is enabled. The best result in each column is bold.

Components				Task-IL performance (%)					
Anchor	CRC	DMEL	Feat.-PGD	Clean	FGSM	PGD-ℓ_∞	PGD-ℓ_2	BIM	RAvg
	✓	✓	✓	94.98	37.01	34.82	40.87	31.98	36.17
✓		✓	✓	94.84	71.88	58.52	93.49	57.61	70.38
✓	✓		✓	94.38	78.05	**72.25**	93.44	71.87	78.90
✓	✓	✓		94.54	78.16	71.90	93.76	71.61	78.86
✓	✓	✓	✓	**95.05**	**78.58**	72.07	**94.18**	**72.46**	**79.32**

4.3 Component Ablation

Component Contributions. Table 2 evaluates each component under the unified Split CIFAR-10 protocol. Removing robust anchoring causes the largest adversarial degradation, while CRC, DMEL, and Feature-PGD provide complementary gains through checkpoint filtering, expert diversity, and feature-level adversarial training. The complete model achieves the highest RobustAvg and the best performance in five of six metrics, with only a slightly lower PGD-ℓ_∞ accuracy than the variant without DMEL.

Local Adversarial-Loss Geometry. Figure 4 visualizes the local adversarial-loss geometry around different checkpoints. Removing robust anchoring produces a sharper and more variable surface, whereas removing CRC yields a smoother landscape but lower adversarial performance. Feature-PGD causes relatively limited geometric changes. Overall, the results are consistent with robust anchoring stabilizing the local adversarial-loss geometry, while the remaining modules provide complementary effects.

5 Conclusion

We presented Robust Expert Collaboration (REC), a replay-free framework for adversarial task-incremental learning. By combining diversified task experts, guarded backbone integration, and class-sensitive checkpoint selection, REC achieves a competitive clean–robustness trade-off on Split CIFAR-10 and Split CIFAR-100.

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

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. *arXiv preprint arXiv:1812.00420*, 2018.

Cheng Chen, Pengpeng Zeng, Yuyu Guo, Lianli Gao, Hengtao Shen, and Jingkuan Song. Janus-LoRA: A balanced low-rank adaptation for continual learning. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://arxiv.org/abs/2605.28495.

Haoran Chen, Ping Wang, Zihan Zhou, Xu Zhang, Zuxuan Wu, and Yu-Gang Jiang. Achieving more with less: Additive prompt tuning for rehearsal-free class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 340–349, 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. doi: 10.1609/aaai.v39i3.32254.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2206–2216, 2020. URL https://proceedings.mlr.press/v119/croce20b.html.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385, 2021.

Yian Deng and Tingting Mu. Understanding and improving ensemble adversarial defense. *Advances in Neural Information Processing Systems*, 36, 2024.

Shibhansh Dohare, J Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Loss of plasticity in deep continual learning. *Nature*, 632(8026): 768–774, 2024.

Yue Duan, Taicai Chen, Lei Qi, and Yinghuan Shi. Divide-and-conquer for enhancing unlabeled learning, stability, and plasticity in semi-supervised continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 583–593, 2025.

Micah Goldblum, Liam Fowl, Soheil Feizi, and Tom Goldstein. Adversarially robust distillation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 3996–4003, 2020.

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

Jiangpeng He, Zhihao Duan, and Fengqing Zhu. CL-LoRA: Continual low-rank adaptation for rehearsal-free class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30534–30544, 2025.

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.

J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell. Overcoming catastrophic forgetting in neural networks. *Proc. of the National Academy of Sciences (PNAS)*, 114(13):3521–3526, 2017.

Jiashuo Li, Shaokun Wang, Bo Qian, Yuhang He, Xing Wei, Qiang Wang, and Yihong Gong. Dynamic integration of task-specific adapters for class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30545–30555, 2025.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.

Meng Lou, Yunxiang Fu, and Yizhou Yu. Scaling continual learning to 300+ tasks with bi-level routing mixture-of-experts. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL `https://arxiv.org/abs/2602.03473`.

Mao-Lin Luo, Yi-Lin Zhang, Zi-Hao Zhou, Yankun Hong, Xialiang Tong, Mingxuan Yuan, Tong Wei, and Min-Ling Zhang. KeepLoRA++: Continual learning with layer-scaled residual gradient adaptation. *arXiv preprint arXiv:2606.16256*, 2026.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *stat*, 1050(9), 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 562–572, 2025.

German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural networks*, 113:54–71, 2019.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. iCaRL: Incremental classifier and representation learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 2001–2010, 2017.

James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11909–11919, 2023.

Yan Wang, Da-Wei Zhou, and Han-Jia Ye. Integrating task-specific and universal adapters for pre-trained model-based class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 806–816, 2025.

Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European conference on computer vision*, pp. 631–648. Springer, 2022a.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 139–149, 2022b.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pp. 3987–3995. Pmlr, 2017.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7472–7482. PMLR, 2019. URL http://proceedings.mlr.press/v97/zhang19p.html.

Yanghao Zhang, Tianle Zhang, Ronghui Mu, Xiaowei Huang, and Wenjie Ruan. Towards fairness-aware adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24746–24755, 2024.

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.

A Additional Implementation Details

A.1 Unified Evaluation and Baseline Reimplementation Protocol

Unified benchmark and evaluation protocol. Split CIFAR-10 contains five disjoint binary tasks in the fixed class order $(4, 2)$, $(7, 6)$, $(0, 3)$, $(5, 8)$, and $(9, 1)$. Training is fully supervised, and Task-IL evaluation uses the task identity to select the corresponding local expert. All numerical entries in Table 1 are produced in our codebase under one controlled protocol; they are not copied directly from the original papers. Unless explicitly noted, every method uses the same CLIP-pretrained ViT-B/16 backbone, 224×224 input resolution, task order, task partitions, train/test split, initialization, optimizer schedule, training budget, and task-aware white-box evaluation. The replay column in Table 1 reports the total number of stored training images; unless a replay budget is listed, no image from a completed task is retained. Native baselines retain their original clean-learning objectives, whereas the aligned variants additionally receive the common post-resize adversarial-training stage specified below.

Baseline status and naming. We re-implement prior continual-learning and adversarial continual-learning methods in the common ViT-based Task-IL framework. For consistency, all baselines augmented with the unified adversarial-training protocol are denoted using the suffix "-AT". Specifically, USP-FSR-AT denotes the FSR component of USP adapted to our fully supervised Task-IL setting with adversarial training; FLAIR-AT denotes our FLAIR-based adversarially trained implementation; DSGD-FS-AT and CaRE-FS-AT denote DSGD-FS and CaRE-FS, respectively, augmented with the same unified adversarial-training protocol. These variants should be interpreted as controlled reimplementation or adaptation baselines rather than as numerical results directly reported in the source papers. The remaining methods are re-implemented under the same backbone, training budget, and attack protocol.

Unified adversarial evaluation. Perturbations are optimized in the unnormalized resized-pixel space, projected onto the corresponding ℓ_∞ or ℓ_2 threat set, clamped to $[0, 1]$, and normalized only before the model forward pass. FGSM, PGD-ℓ_∞, and BIM use $\epsilon = 8/255$; the iterative attacks use step size $2/255$ and 10 steps. PGD-ℓ_2 uses radius $128/255$, step size $15/255$, and 10 steps. Each attack targets the active task-local expert. We macro-average accuracy across tasks, and define RobustAvg as the arithmetic mean of FGSM, PGD-ℓ_∞, PGD-ℓ_2, and BIM accuracy, excluding clean accuracy. The strong audit varies the number of PGD steps and random restarts and additionally evaluates task-local binary AutoAttack Croce & Hein (2020).

Training configuration. DMEL uses the current expert and two noisy copies, five collaboration epochs, image-space PGD-7 with $\epsilon = 8/255$ and step size $2/255$, and feature-space PGD-3 with relative radius $0.05\|z\|_2$. The Task-0 robust-anchor stage uses 12 epochs, including two clean-warmup epochs and a six-epoch perturbation curriculum. Its physical batch size is 4 with 16-step gradient accumulation. The subsequent robust-integration stage uses 8 epochs for Task 0 and 10 epochs for later tasks, with PGD-10, the same physical and effective batch sizes, and cosine learning-rate decay. During robust integration, the early backbone, the last four transformer blocks together with the final

Table 3: Status of the methods evaluated under the unified protocol. The suffix "-AT" denotes adaptation with the unified adversarial-training protocol.

Method	Status under our evaluation protocol
REC	Proposed method
USP-FSR-AT	USP-FSR adaptation + unified AT
FLAIR-AT	FLAIR-based implementation + unified AT
DSGD-FS-AT	DSGD-FS + unified AT
CaRE-FS-AT	CaRE-FS + unified AT
CaRE-FS, DSGD-FS, DER-ViT, DRIFT-TIL	Re-implemented without the added AT stage

normalization layer, and the current expert use learning rates 10^{-6}, 10^{-5}, and 5×10^{-4}, respectively. CRC does not introduce an additional optimization stage; instead, it evaluates candidate checkpoints generated during robust integration on a held-out gating split and performs checkpoint selection and rollback according to the criteria described below.

Implementation form of the RASC robust-integration objective. The robust-integration stage implements the RASC objective as

$$\begin{aligned}\mathcal{L}_{\mathrm{RASC}}^{\mathrm{impl}} = \lambda_c \,\mathrm{CE}(p(x), y) + \lambda_a \,\mathrm{CE}(p(x^{\mathrm{adv}}), y) + \lambda_{\mathrm{tr}}\, T^2 D_{\mathrm{KL}}(p_T(x) \| p_T(x^{\mathrm{adv}})) \\ + \lambda_{\mathrm{inv}}[1 - \cos(z(x), z(x^{\mathrm{adv}}))] + \lambda_{\mathrm{teach}}\, \mathcal{L}_{\mathrm{teacher}} + \lambda_{\mathrm{old}}\, \mathcal{L}_{\mathrm{old}},\end{aligned} \tag{16}$$

where $(\lambda_c, \lambda_a, \lambda_{\mathrm{tr}}, \lambda_{\mathrm{inv}}, \lambda_{\mathrm{teach}}, \lambda_{\mathrm{old}}) = (1, 1, 0.5, 0.2, 1, 1)$ and $T = 2$. Here $\mathcal{L}_{\mathrm{teacher}}$ is the mean clean/adversarial cosine distance between the current and frozen-teacher CLS features, and $\mathcal{L}_{\mathrm{old}}$ averages temperature-scaled KL divergence over the frozen historical experts on clean and adversarial inputs. This implementation corresponds to the robust-integration objective described by RASC in the main text. CRC is applied only after candidate checkpoints are generated and therefore does not introduce a separate training loss.

Validation gates and rollback. CRC constructs a held-out gating split $\widetilde{D}_t$ by reserving 10% of the current task training set D_t for checkpoint selection. All CRC statistics are computed exclusively on $\widetilde{D}_t$. Let A_{clean} denote clean accuracy on the gating split, A_{adv} denote adversarial accuracy under PGD-ℓ_∞, and R_c^{adv} denote the adversarial recall of class c. We define the worst-class adversarial recall as

$$R_{\min}^{\mathrm{adv}} = \min_c R_c^{\mathrm{adv}}.$$

A candidate checkpoint is considered valid only if

$$\begin{aligned} A_{\mathrm{clean}} &\geq \tau_{\mathrm{clean}}, \\ A_{\mathrm{adv}} &\geq \tau_{\mathrm{adv}}, \\ R_{\min}^{\mathrm{adv}} &\geq \tau_{\mathrm{recall}}. \end{aligned} \tag{17}$$

Among all candidates satisfying these gating conditions, checkpoints are selected according to their validation performance in adversarial accuracy, clean accuracy, and historical decision consistency. If no candidate satisfies all gating conditions, the algorithm restores both the shared backbone and the current expert to the post-DMEL fusion state. CRC is therefore a validation-time checkpoint-selection and rollback mechanism rather than an additional optimization objective. The test set is used only for final evaluation and is never accessed during checkpoint selection.

The Task-0 robust-anchor stage uses a separate gate: clean accuracy at least 92%, minimum clean class recall at least 80%, and PGD-10 accuracy at least 1%. It selects valid candidates by PGD and then clean accuracy; if no epoch is valid, it restores the pre-anchor clean model. Historical experts and all teacher snapshots remain frozen throughout their corresponding stages.

Continual-learning metrics. Let $A_{s,t}$ denote accuracy on task t after learning stage s, computed separately for clean and PGD-ℓ_∞ inputs. For T tasks, final average accuracy is $\frac{1}{T}\sum_{t=1}^{T} A_{T,t}$, and average incremental accuracy is $\frac{1}{T}\sum_{s=1}^{T}\frac{1}{s}\sum_{t=1}^{s} A_{s,t}$. Backward transfer is $\frac{1}{T-1}\sum_{t=1}^{T-1}(A_{T,t} - A_{t,t})$, and forgetting is $\frac{1}{T-1}\sum_{t=1}^{T-1}(\max_{s\in\{t,\ldots,T-1\}} A_{s,t} - A_{T,t})$. Old-task accuracy averages tasks 1 through $T-1$, whereas new-task accuracy is $A_{T,T}$.

B Robustness and Reproducibility Diagnostics

The following two tables separate attack-side diagnostics from optimization-seed reproducibility so that each audit can be read independently.

B.1 Strong-attack and final-evaluation diagnostics

Increasing PGD from 10 steps and one restart to 40 steps and five restarts reduces accuracy from 72.07% to 70.11%, while task-local binary AutoAttack obtains 68.51% (Table 4(a)). The monotonic reduction confirms that stronger optimization finds additional errors, but the 1.96-point PGD drop is substantially smaller than the gap between REC and the native baselines. This pattern is inconsistent with robustness that exists only under a short, single-start attack, although neither sweep constitutes a formal certificate.

The final unified evaluation reveals a complementary limitation. REC maintains high clean accuracy (95.05%) while achieving 72.07% PGD-ℓ_∞ accuracy and a RobustAvg of 79.32%. Although stronger attacks further reduce performance, the overall degradation remains moderate relative to the single-start PGD-10 result. These results indicate that REC preserves a favorable clean–robustness trade-off under the unified evaluation protocol, while leaving room for further improvement under stronger adversarial optimization.

Table 4: Robustness diagnostics for REC on Split CIFAR-10 under the unified post-resize evaluation protocol.

(a) Strong-attack audit

Attack	Steps	R	Acc. (%)
PGD-ℓ_∞	10	1	72.07
PGD-ℓ_∞	10	5	72.01
PGD-ℓ_∞	20	1	70.55
PGD-ℓ_∞	20	5	70.47
PGD-ℓ_∞	40	1	70.20
PGD-ℓ_∞	40	5	70.11
AutoAttack	–	–	68.51

(b) Final unified evaluation

Metric	Accuracy (%)
Clean	95.05
FGSM	78.58
PGD-ℓ_∞	72.07
PGD-ℓ_2	94.18
BIM	72.46
RobustAvg	79.32

Note. R denotes the number of random restarts. Panel (a) evaluates the seed-1993 REC checkpoint under increasingly strong PGD-ℓ_∞ configurations and task-local binary AutoAttack. Panel (b) reports the final seed-1993 Task-IL results under the same unified evaluation protocol as Table 1. RobustAvg is the arithmetic mean of FGSM, PGD-ℓ_∞, PGD-ℓ_2, and BIM accuracy.

B.2 Three-Seed Reproducibility

Across three REC optimization seeds with the class order fixed, every standard deviation is below one percentage point (Table 5). This establishes low within-method seed variability for the reported order without claiming invariance to alternative task orders.

Table 5: Three-seed reproducibility of REC under unified post-resize 224×224 Task-IL evaluation.

Metric	1993	2024	3407	Mean $\pm$ Std.
Clean	95.05	94.03	94.31	94.46 ± 0.43
FGSM	78.58	77.66	77.56	77.93 ± 0.46
PGD-ℓ_∞	72.07	71.68	70.52	71.42 ± 0.66
PGD-ℓ_2	94.18	93.01	93.52	93.57 ± 0.48
BIM	72.46	72.05	70.96	71.82 ± 0.63
RobustAvg	79.32	78.60	78.14	$\mathbf{78.69 \pm 0.49}$

Note. Values are percentages. The final column reports the population mean and standard deviation over optimization seeds 1993, 2024, and 3407 with the task order fixed.

C Extended Continual and Mechanistic Analysis

C.1 Stage-by-Task Robust Retention

Figure 5 presents the lower-triangular stage-by-task evaluation history of REC. The clean matrix remains consistently high after the first stage. Under PGD-ℓ_∞, several historical tasks maintain or slightly improve their robustness after later updates, indicating that robust performance on earlier tasks is largely preserved throughout the sequence. In contrast, the lower final diagonal entry shows that the newly introduced Task 4 remains substantially more difficult, suggesting a remaining limitation in robust plasticity on the newest task rather than broad degradation across historical tasks.

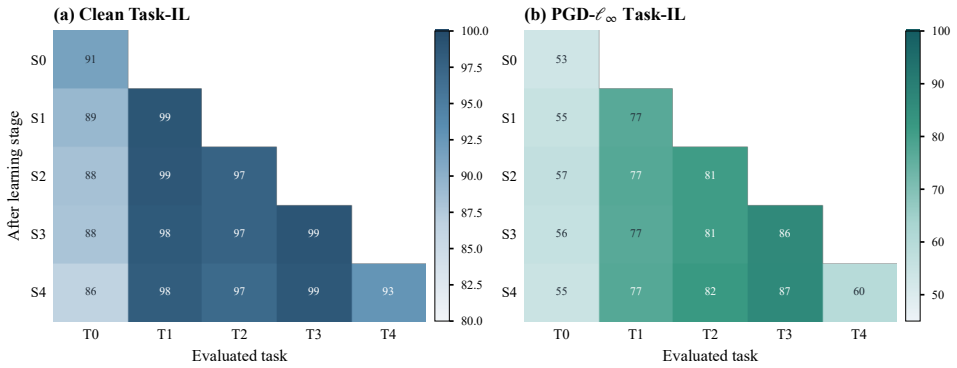


Figure 5: Stage-by-task Task-IL accuracy matrices for REC. Row S_s evaluates all observed tasks after stage s. **(a)** Clean evaluation. **(b)** Post-resize PGD-ℓ_∞-10 evaluation.

C.2 Layer-wise Adversarial Representation Drift

For ViT block ℓ, let $z_\ell(x)$ denote its CLS representation. We measure relative displacement $100\,\mathbb{E}[\|z_\ell(x^{\mathrm{adv}}) - z_\ell(x)\|_2/(\|z_\ell(x)\|_2 + \epsilon)]$ and directional displacement $100\,\mathbb{E}[1 - \cos(z_\ell(x^{\mathrm{adv}}), z_\ell(x))]$ over the full Split CIFAR-10 test set. Figure 6 shows that full REC limits its displacement in early layers and concentrates the residual change in later task-discriminative blocks. Removing the anchor increases block-12 relative displacement from 28.31% to 111.46% and the scaled cosine distance from 3.90 to 46.74, accompanying a PGD decrease from 72.04% to 34.66%. Removing CRC leaves displacement closer to full REC but lowers PGD accuracy to 58.51%, supporting the distinct checkpoint-selection role of CRC.

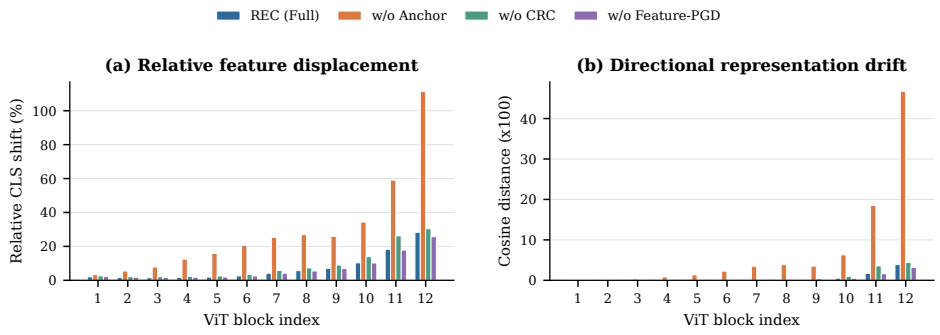


Figure 6: Layer-wise sensitivity to post-resize PGD-ℓ_∞-10. **(a)** Relative ℓ_2 displacement between clean and adversarial CLS representations. **(b)** Cosine distance multiplied by 100.

C.3 Robust-Anchor Optimization and CRC Selection Dynamics

Figure 7 separates the trajectory-shaping role of robust-anchor optimization from the validation-time checkpoint-selection role of CRC. Once the clean warmup ends, anchor PGD accuracy rises from below 10% to above 50%, but only a subset of epochs satisfies the anchor-selection criteria. CRC

exhibits task-dependent selection behavior: relative to epoch 1, the selected gate-valid checkpoint improves PGD accuracy by 26.6, 26.8, 9.4, 12.0, and 28.8 points on Tasks 0–4, respectively. The worst-class recall change stays within 0.6 points on Tasks 1–4 but decreases by 8.9 points on Task 0, which remains the principal difficult case. These results illustrate that CRC filters candidate checkpoints produced during robust integration rather than directly modifying the optimization trajectory.

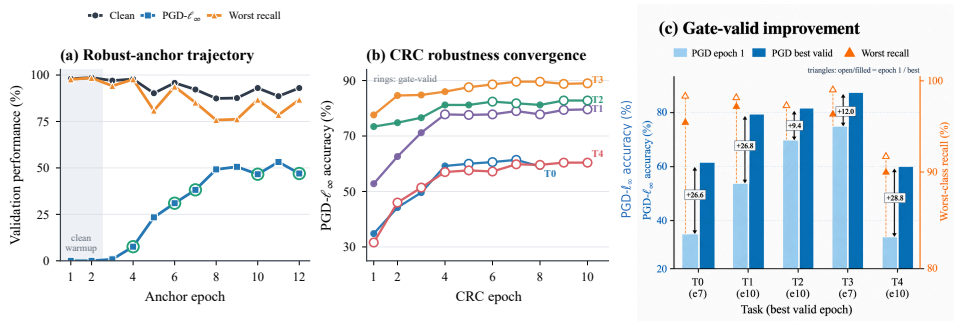


Figure 7: Robust-anchor optimization and CRC checkpoint-selection dynamics. **(a)** Anchor validation trajectory; ringed markers denote checkpoints satisfying the anchor-selection criteria, and the shaded interval indicates clean warmup. **(b–c)** Task-wise CRC checkpoint-selection behavior; highlighted cells and markers denote gate-valid candidates.

D Matched-Seed Comparison with USP-FSR-AT

We compare REC and USP-FSR-AT under three matched optimization seeds while fixing the class order to $(4, 2)$, $(7, 6)$, $(0, 3)$, $(5, 8)$, and $(9, 1)$. All runs use the same post-resize 224×224 Task-IL evaluation and the attack settings described in Appendix A. RobustAvg is the arithmetic mean of FGSM, PGD-ℓ_∞, PGD-ℓ_2, and BIM accuracy. The reported standard deviations are population standard deviations across the three seeds.

Table 6: Matched-seed comparison of REC and USP-FSR-AT on Split CIFAR-10.

(a) Three-seed summary (%)

Method	Clean	FGSM	PGD-ℓ_∞	PGD-ℓ_2	BIM	RobustAvg
REC (Ours)	**94.46 ± 0.43**	**77.93 ± 0.46**	71.42 ± 0.66	**93.57 ± 0.48**	**71.82 ± 0.63**	**78.69 ± 0.49**
USP-FSR-AT	94.38 ± 0.72	77.83 ± 0.31	**71.80 ± 0.08**	93.56 ± 0.78	71.40 ± 0.15	78.65 ± 0.22

(b) Matched-seed results (%)

Method	Seed	Clean	FGSM	PGD-ℓ_∞	PGD-ℓ_2	BIM	RobustAvg
REC (Ours)	1993	95.05	78.58	72.07	94.18	72.46	79.32
	2024	94.03	77.66	71.68	93.01	72.05	78.60
	3407	94.31	77.56	70.52	93.52	70.96	78.14
USP-FSR-AT	1993	93.44	77.54	71.82	92.53	71.51	78.35
	2024	95.19	78.25	71.69	94.42	71.19	78.89
	3407	94.50	77.69	71.88	93.73	71.51	78.70

Values are percentages. Means and population standard deviations are computed over optimization seeds 1993, 2024, and 3407 with the task order fixed. Bold denotes the higher three-seed mean between the two methods.

Table 6 shows that the two methods are closely matched after averaging over optimization seeds. REC is higher by 0.09 points in clean accuracy, 0.11 points under FGSM, 0.42 points under BIM, and 0.04 points in RobustAvg, whereas USP-FSR-AT is higher by 0.37 points under PGD-ℓ_∞. PGD-ℓ_2 differs by only 0.01 points. The ranking changes across seeds, so these runs support a conclusion of comparable aggregate robustness rather than consistent statistical superiority by either method.

The paired differences in Table 7 change sign across seeds. In particular, REC leads on seed 1993, while USP-FSR-AT obtains a higher RobustAvg on seeds 2024 and 3407. We therefore use the multi-seed results to quantify optimization variability and do not treat the single-seed margin in the main table as evidence of seed-independent dominance.

Table 7: Paired seed-wise differences between REC and USP-FSR-AT. Positive values favor REC; negative values favor USP-FSR-AT. Values are percentage points.

Seed	Clean	FGSM	PGD-ℓ_∞	PGD-ℓ_2	BIM	RobustAvg
1993	+1.61	+1.04	+0.25	+1.65	+0.95	+0.97
2024	-1.16	-0.59	-0.01	-1.41	+0.86	-0.29
3407	-0.19	-0.13	-1.36	-0.21	-0.55	-0.56
Mean paired difference	+0.09	+0.11	-0.37	+0.01	+0.42	+0.04

D.1 Task-Wise Strong-Attack Results for USP-FSR-AT

Table 8: Task-wise robustness of USP-FSR-AT for the two newly completed fixed-order seeds. The AutoAttack column uses task-local binary AutoAttack. The final row is the macro-average across tasks.

	Seed 2024		Seed 3407		Mean	
Task	PGD-10	AA	PGD-10	AA	PGD-10	AA
T0	45.15	40.60	44.25	40.45	44.70	40.53
T1	74.55	71.50	75.90	73.45	75.23	72.48
T2	82.30	81.00	82.75	80.90	82.53	80.95
T3	88.15	86.65	88.10	86.25	88.13	86.45
T4	68.30	65.25	68.40	64.90	68.35	65.08
Macro-average	71.69	69.00	71.88	69.19	71.79	69.10

The two USP-FSR-AT runs exhibit similar task-wise behavior. The macro PGD-ℓ_∞ scores differ by 0.19 points and the AutoAttack scores differ by 0.19 points. AutoAttack reduces the two-seed macro accuracy from 71.79% to 69.10%, while preserving the task ranking. Task 0 remains the most difficult and Task 3 the strongest under both attacks.

E Efficiency and Resource Cost

We report resource measurements for REC and USP-FSR-AT, the strongest matched baseline considered in the seed analysis. Stage time is measured by the training runners and includes stage-local training, clean evaluation, and checkpoint writing, but excludes the final unified post-resize adversarial evaluation. All reported timing and peak-memory measurements were collected on NVIDIA GeForce RTX 3090 GPUs under the same-device profiling setting. Active-path parameter counts are computed from the modules used for one task-aware forward pass after Task 4.

Table 9: Training time and peak-memory profile. Stage times are in minutes. Sum is the sum of the five recorded stage times; wall time is the end-to-end elapsed time. Peak GPU denotes the maximum GPU memory observed during training under the same-device profiling setting.

Method	T0	T1	T2	T3	T4	Sum (h)	Wall (h)	Peak GPU (GiB)
REC (Ours)	360.45	283.71	283.55	286.49	301.99	25.27	25.72	5.60
USP-FSR-AT	303.38	120.28	120.46	120.61	120.78	13.09	13.11	2.56

Table 10: Unified model-footprint and deployment profile after Task 4. Parameter counts are in millions. Stored parameters count the full retained model, whereas active-path parameters count only the modules used for one task-aware prediction.

Method	Trainable T0	Trainable T1–T4	Stored after T4	Active path	Replay imgs.	Temporary experts at inference
REC (Ours)	86.197	0.390	87.755	86.197	0	Not used
USP-FSR-AT	86.590	86.590	88.148	86.590	0	Not used

REC incurs a larger wall-clock training cost because robust expert construction, guarded backbone integration, and checkpoint feasibility checks require additional adversarial optimization. Its efficiency

advantage is instead concentrated in incremental update size: after Task 0, REC optimizes only the current 0.390M-parameter task expert, whereas USP-FSR-AT updates approximately 86.590M parameters at every subsequent stage. We therefore describe REC as parameter-efficient during incremental updates, not as faster in end-to-end training.

Table 10 reports storage, update size, replay use, and active-path model footprint in one view. DMEL creates multiple temporary experts only during task-expert construction. Uniform fusion maps these branches to one compact task expert before RASC and CRC, and the temporary branches are not retained in the final checkpoint. Consequently, DMEL does not multiply the number of task experts activated for a test sample. After Task 4, REC stores 87.755M parameters in total but activates only 86.197M parameters for one task-aware prediction; USP-FSR-AT activates 86.590M parameters. REC still stores one compact expert per learned task, so total checkpoint storage grows with the task sequence even though Task-IL inference activates only the expert selected by the known task identity.

F Scope and Limitations

The experiments evaluate replay-free Task-IL, where the task identity is available at inference and selects the corresponding local expert. The results therefore do not establish Class-IL performance when the task identity is unknown. Optimization-seed experiments keep one class order fixed; they measure within-order training variability rather than robustness to alternative task orders.

CRC uses a deterministic held-out subset of the current training task for checkpoint selection and never accesses the test labels during training. Its constraints reduce the frequency of class-collapsed validation checkpoints, but they do not guarantee robustness on unseen examples. Likewise, the PGD sweep and task-local AutoAttack audit are empirical stress tests rather than formal robustness certificates.

The stage-by-task analysis identifies a remaining plasticity limitation: REC preserves historical adversarial performance but performs less strongly on the final newly introduced task. REC also has a higher end-to-end training cost than USP-FSR-AT. These results delimit the present contribution: REC targets robust retention and parameter-efficient incremental updates. Reducing robust-training time and improving newest-task plasticity remain important directions for future work.