Improving Sustainability of Adversarial Examples in Class-Incremental Learning

Taifeng Liu, Xinjing Liu*, Liangqiu Dong, Yang Liu, Yilong Yang, Zhuo Ma*

School of Cyber Engineering, Xidian University, China
tfliu@gmx.com, liuxinjing_j@163.com, liangqiu.dong@stu.xidian.edu.cn,
bcds2018@foxmail.com, yilongyang@xidian.edu.cn, mazhuo@mail.xidian.edu.cn

Abstract

Current adversarial examples (AEs) are typically designed for static models. However, with the wide application of Class-Incremental Learning (CIL), models are no longer static and need to be updated with new data distributed and labeled differently from the old ones. As a result, existing AEs often fail after CIL updates due to significant domain drift. In this paper, we propose SAE to enhance the sustainability of AEs against CIL. The core idea of SAE is to enhance the robustness of AE semantics against domain drift by making them more similar to the target class while distinguishing them from all other classes. Achieving this is challenging, as relying solely on the initial CIL model to optimize AE semantics often leads to overfitting. To resolve the problem, we propose a Semantic Correction Module. This module encourages the AE semantics to be generalized, based on a visual-language model capable of producing universal semantics. Additionally, it incorporates the CIL model to correct the optimization direction of the AE semantics, guiding them closer to the target class. To further reduce fluctuations in AE semantics, we propose a Filtering-and-Augmentation Module, which first identifies non-target examples with target-class semantics in the latent space and then augments them to foster more stable semantics. Comprehensive experiments demonstrate that it SAE outperforms baselines by an average of 31.28% when updated with a 9-fold increase in the number of classes.

Code — https://github.com/Jupiterliu/SAE
Extended version — https://arxiv.org/abs/2511.09088

Introduction

Adversarial examples (AEs) pose a significant threat to machine learning models, especially in safety-critical applications like autonomous driving, healthcare, etc. (Badjie, Cecilio, and Casimiro 2024). These attacks work by perturbing input data to deceive models into making incorrect predictions. Current AEs are typically designed for static models (Pelekis et al. 2025). However, with the advancement of Class-Incremental Learning (CIL) (Zhou et al. 2024b,a; Zhang et al. 2025), models are no longer static, but sequentially updated with examples distributed and labeled differently from the old ones. The dynamic nature of CIL causes AEs generated on the old model to become ineffective after CIL updates. As demonstrated in Figure 1, a small update with just 30 classes on a ResNet-32 can lead to a significant reduction in attack success rate when evaluated with state-of-the-art (SOTA) AEs (Dong et al. 2018; Li, Ma, and Jiang 2025; Xu et al. 2025). Given the broad applicability of CIL, ensuring the *sustainability* of AEs in CIL scenarios is crucial, prompting the need to explore why AEs fail in CIL.

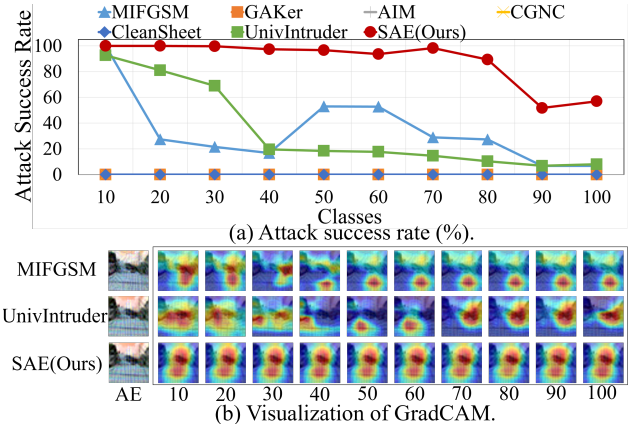

Figure 1: Attack success rate and GradCAM of different targeted adversarial attacks against CIL. The X-axis denotes the number of learned classes in CIL, with the model architecture being ResNet-32 on CIFAR-100.

Domain Drift Causes AE Failures. Targeted AEs essentially add perturbations to input data, driving examples from the source-class domain across the model's decision boundary into the target-class domain (Dong et al. 2018; Li, Ma, and Jiang 2025) [1]. However, current AEs are typically designed for static models. When a CIL model is updated with new-class examples, the domains of all previous learned classes undergo significant drift (Masana et al. 2022; Li et al. 2025). This domain drift alters both the direction and magnitude of the perturbations required to shift inputs toward the target domain. As a result, most AEs either mislead inputs into incorrect classes or deteriorate into benign noise,

*Corresponding author.

[1] Untargeted AEs are not considered in this work, as sufficient perturbation can effectively mislead CIL models.

rendering them ineffective. Although advanced adversarial attacks attempt to improve transferability by embedding semantic information of the target class (Li, Ma, and Jiang 2025; Fang et al. 2024; Sun et al. 2024; Ge et al. 2024; Xu et al. 2025), they still struggle to overcome the significant domain drift affecting old classes. The green line in Figure 1(a) illustrates a typical semantic-based AE, which maintains an attack success rate above 20% for no more than 30 incremental classes. Figure 1(b) shows GradCAM (Selvaraju et al. 2017) visualizations of various AEs in the CIL setting. As the number of learned classes increases, the effectiveness of current attacks gradually diminishes, with the regions contributing to the target class shrinking over time.

In this paper, we improve the sustainability of adversarial examples in CIL by proposing SAE. The core idea of SAE is to enhance the robustness of AE semantics against domain drift by making them similar to the target class while distinguishing them from all other classes. However, realizing this idea faces two main challenges: 1) AEs tend to overfit if perturbations are optimized based solely on the gradients of the initial CIL model; 2) Non-target examples unintentionally contain target-class semantics, leading to fluctuations in AE semantics. For example, images labeled as 'bicycle' may also contain 'roads' when 'roads' is the target class. To address these issues, we design two modules: the Semantic Correction Module and the Filtering-and-Augmentation Module. The first module encourages generalized AE semantics by incorporating a visual-language model that provides universal target-class semantics as an 'anchor'. Additionally, it uses the gradients of the CIL model to guide the optimization of AE semantics, ensuring consistency with the target class throughout the CIL process. The second module detects examples with confusing semantics by calculating the cosine similarity between the non-target class and target-class examples in the latent space. The remaining examples with low similarities are further augmented to promote more stable and generalized semantics.

We highlight our contributions as follows:

- To the best of our knowledge, we are the first to investigate the sustainability of adversarial attacks under the setting of class-incremental learning.
- We propose SAE to improve the sustainability of AEs in CIL, which enhances AE robustness by making their semantics similar to the target class while distinguishing them from all other classes.
- To prevent AE semantics from overfitting, we propose a Semantic Correction Module that promotes generalized AE semantics towards the target class using a visual-language model and corrects the optimization direction based on the CIL model.
- We propose a Filtering-and-Augmentation Module that removes examples with confusing semantics in the latent space and augments the remaining examples to ensure stable and generalized AE semantics.
- Extensive experiments show that SAE significantly outperforms existing approaches in terms of sustainability, improving the average attack success rate by 31.28% after CIL with a 9-fold increase in the number of classes.

Related Works

Class-Incremental Learning

Class-Incremental Learning is a specific form of incremental learning (also referred to as continuous learning (De Lange et al. 2022) or lifelong learning (Chen and Liu 2018)) (Wang et al. 2024), where the model learns to classify new classes incrementally. CIL allows users to update the model once new-class data is available. This makes CIL more adaptable and suitable for dynamic environments (Zhou et al. 2024c; Ashtekar, Zhu, and Honavar 2025), making it a prominent and extensively studied paradigm in incremental learning.

Current researches of CIL mainly focus on reducing catastrophic forgetting on old-class data. There are mainly five types of CIL methods: 1) Finetune is a typical baseline in CIL, which only utilizes the cross-entropy loss in new tasks to update the model while ignoring former tasks. It suffers from severe forgetting of former tasks; 2) Replay-based methods, such as Replay (Ratcliff 1990) and RMM (Liu, Schiele, and Sun 2021), which store interleave old-class data with new data; 3) Knowledge distillation leverages soft targets from previous models as regularization, such as iCaRL (Rebuffi et al. 2017a), PodNet (Douillard et al. 2020), and LKD (Gao et al. 2025a); 4) Dynamic network, such as DER (Buzzega et al. 2020), Foster (Wang et al. 2022a), MEMO (Zhou et al. 2022), L2P (Wang et al. 2022b), and TagFex (Zheng et al. 2025), which dynamically expand the model architecture or adjust parameters to accommodate new tasks; 5) Model rectification aims to reduce the biased prediction of incremental learners, such as BiC (Wu et al. 2019), WA (Zhao et al. 2020), and MoAL (Gao et al. 2025b).

Targeted Transferable Adversarial Attack

Targeted transferable adversarial attacks represent a more challenging scenario where the adversary aims to mislead models into specific incorrect classes. Existing targeted attacks can be broadly categorized into *Iterative Attacks* and *Generative Attacks* (Badjie, Cecilio, and Casimiro 2024; Li, Ma, and Jiang 2025). Iterative attacks craft AEs based on logit-oriented loss functions, often leveraging gradients and surrogate models to enhance transferability. Traditional targeted AEs like fast gradient sign method (FGSM) (Goodfellow, Shlens, and Szegedy 2014) and the momentum iterative FGSM (MIFGSM) (Dong et al. 2018) exploit the model's gradients to maximize the loss. Recent iterative attacks focus more on the transferability of attacks to models with different architectures by extracting more robust features, such as CFM (Byun et al. 2023), CleanSheet (Ge et al. 2024), UnivIntruder (Xu et al. 2025), and LTT (Weng, Luo, and Li 2025). Generative attacks have gained popularity recently due to better transferability, which produces AEs by generative models, such as TTP (Naseer et al. 2021), TTAA (Wang et al. 2023; Sun et al. 2023), CGNC (Fang et al. 2024), GAKer (Sun et al. 2024), AdvDiffVLM (Guo et al. 2024), and AIM (Li, Ma, and Jiang 2025), which leverage abundant semantic information from the latent feature space. Although recent AEs have demonstrated strong transferability across models, they have not considered the impact of catastrophic forgetting in CIL.

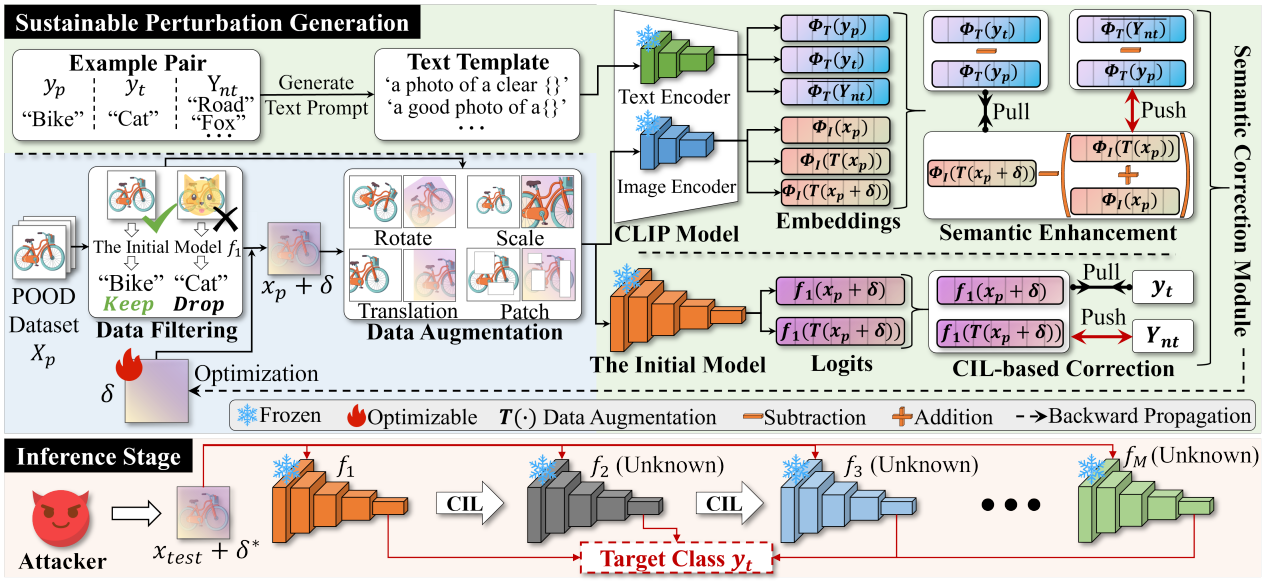

Figure 2: The overview of SAE.

Approach

Problem Formulation

Class-Incremental Learning. In CIL, a model is trained sequentially on M tasks, with each task introducing a set of new classes. Let $f_i(\cdot)$ represent the model trained on the i-th task, where $i \in 1, 2, ..., M$. The model is trained incrementally such that it learns the new classes in each task while maintaining its ability to classify all previously learned classes.

Adversary's Capability & Goals. The adversary can access the initial CIL model $f_1(\cdot)$ and the complete set of labels $\{y_t, Y_{nt}\}$ in the CIL process. y_t represents the target class, and Y_{nt} includes all other classes learned in CIL. Importantly, the adversary has no access to the CIL training set, nor any information regarding the training process used to update the model. To perform an attack, the adversary who knows CIL label can easily access public and label-mismatched data, i.e., public out-of-distribution (POOD) datasets, without knowing the distribution of the training data. Moreover, the adversary has access to a public pre-trained visual-language model for semantic extraction. Given a test dataset X_{test} and any model $f_i(\cdot)$ within the CIL process, the adversary aims to generate a universal perturbation δ: For any image $x \in X_{test}$, the CIL model $f_i(\cdot)$ misclassifies into a specific target class y_t, when applying δ on x. The optimization objective is formally expressed as:

$$\delta^* = \arg\min_{\delta} \mathbb{E}_{x \in X_{test}}[\mathcal{L}(f_i(x+\delta), y_t)], \forall i \geq 1 \quad \text{s.t.} \quad ||\delta|| \leq \epsilon \tag{1}$$

where $\mathcal{L}(\cdot)$ is the loss function. δ is constrained by an l_∞-norm perturbation budget ϵ to ensure stealthy.

Overview of SAE

Based on the core idea of semantic generalization, we design the SAE framework, as illustrated in Figure 2. On the above is sustainable perturbation generation, which consists of two main modules: 1) Filtering-and-Augmentation (blue area) to address semantic fluctuation by detecting examples with confusing semantics; 2) Semantics Correction Module (green area) to optimize adversarial perturbations, ensuring sustainable target-class semantics. At the inference stage (pink area), the adversarial perturbation, once optimized, is applied to any updated CIL model that provides black-box access, with the goal of misclassifying the perturbed examples into the target class y_t. For clarity, in the following, we first introduce the optimization of AEs, followed by data filtering and augmentation.

Semantic Correction Module

This part aims to generate universal semantics to optimize AEs, based on the CIL model. CLIP, a pre-trained visual-language model, is widely considered to be capable of extracting and generating stable semantic representations for specific classes (Radford et al. 2021) [2]. Previous attacks (Xu et al. 2025; Fang et al. 2024) also demonstrate that CLIP can generate stable targeted semantics. Therefore, we apply CLIP to provide universal semantic information, which aligns images and text in a shared embedding space.

CLIP-Based Semantic Enhancement. To provide abundant semantics of non-target class and make target-class semantics distinguishable, we access a POOD dataset, denoted as $D_p = \{X_p, Y_p\}$. For any $y_p \in Y_p$, $y_p \neq y_t$ and $y_p \notin Y_{nt}$. In CLIP, the mapping of images and texts is

[2] CLIP is trained on billions of image-text pairs, making it particularly well-suited for zero-shot classification.

Algorithm 1: Pseudocode of Perturbation Optimization

Require: initial CIL model $f_1(\cdot)$, CLIP encoders $\{\Phi_T(\cdot), \Phi_I(\cdot)\}$, POOD dataset D_p, target class y_t, non-target classes Y_{nt}, constraint ϵ, Filtering Function $F_{\text{filter}}(\cdot)$, Augmentation Function $F_{\text{aug}}(\cdot)$.
Ensure: Optimized perturbation δ.
1: Initialize $\delta \sim$ Gaussian(0, 1);
2: **for** each example $x_p, y_p \in D_p$ **do**
3: **while** $F_{\text{filter}}(x_p) > \sigma$ **do**
4: $\hat{x}_p = F_{aug}(x_p)$;
5: $x'_p = F_{aug}(x_p + \delta)$;
6: $D_t = \Phi_T(y_t) - \Phi_T(y_p)$;
7: $D_{nt} = \overline{\Phi_T(Y_{nt})} - \Phi_T(y_p)$;
8: $D_{adv} = \Phi_I(x'_p) - \Phi_I(x_p) - \Phi_I(\hat{x}_p)$;
9: $\text{sim}_{pos} = \frac{D_{adv} \cdot D_t}{||D_{adv}|| \, ||D_t||}$;
10: $\text{sim}_{neg} = \frac{D_{adv} \cdot D_{nt}}{||D_{adv}|| \, ||D_{nt}||}$;
11: $\mathcal{L} = \mathcal{L}_{\text{CLIP}} + \mathcal{L}_{\text{Surr}}$; # Referring Eq. 2 and Eq. 3
12: $\delta \leftarrow \delta - \alpha \bigtriangledown_\delta \mathcal{L}$;
13: $\delta \leftarrow \text{clamp}(\delta, \epsilon)$;
14: **end while**
15: **end for**

realized by two encoders, i.e., the text encoder $\Phi_T(\cdot)$ and the image encoder $\Phi_I(\cdot)$, which accept image or text input and obtain embeddings. Inspired by (Xu et al. 2025), we calculate the target (D_t), non-target (D_{nt}), and adversarial (D_{adv}) semantic directions based on the embeddings of the image-text pairs extracted by corresponding encoders (Line 6-8 in Algorithm 1). Then, SAE calculates two types of similarity: 1) The positive similarity between the perturbed examples ($x_p + \delta$) and the target-class text y_t (pull); 2) The negative similarity between the perturbed examples and the non-target class texts Y_{nt} (push) [3]. Notably, for each y_p, the similarities need to be computed individually, and the optimization should iterate $|Y_p|$ times. Figure 2 illustrates optimization for a single y_p. Based on the similarities, we define $\mathcal{L}_{\text{CLIP}}$ for optimizing δ:

$$\mathcal{L}_{\text{CLIP}} = -log(\frac{\text{sim}_{pos}}{\text{sim}_{pos}+\text{sim}_{neg}}) \tag{2}$$

That is, $\mathcal{L}_{\text{CLIP}}$ guides the AE to align with the target class's semantic embeddings while ensuring they are distant from the semantics of non-target classes. We detail the computation of sim_{pos}, sim_{neg} in Line 6-10 of Algorithm 1. $\overline{\Phi_T(Y_{nt})}$ represents the average of all the embeddings of non-target classes, and its size is the same as that of $\Phi_T(y_p)$.

CIL-Based Correction. Only the static semantics provided by CLIP are hard to sustain due to the semantic drift. Thus, we further refine them using gradients from the initial model. It comes from the fact that guidance from the initial model remains effective due to the preserved gradient through distillation or orthogonal projection in CIL (Zhou et al. 2024b). Based on $f_1(\cdot)$, we obtain logits of all perturbed examples and then compute the Binary Cross-Entropy (BCE) loss between the logits and the target class y_t:

$$\mathcal{L}_{\text{Surr}} = -log(p_{y_t}) - \textstyle\sum_{y_{nt} \in Y_{nt}} log(1 - p_{y_{nt}}) \tag{3}$$

where p_{y_t} and $p_{Y_{nt}}$ represent the predicted probability of the target class and non-target classes made by $f_1(\cdot)$, respectively. $\mathcal{L}_{\text{Surr}}$ ensures that the adversarial perturbations generated based on the surrogate model not only mislead the model prediction but also help improve the sustainability of the attack across evolving models in the CIL process.

Filtering-and-Augmentation Module

To reference confusing semantics, we first gather several typical examples of class y_t to obtain embeddings that represent the target class. These examples can be collected from the POOD datasets, denoted as X_c. Each example in the dataset X_p is first filtered with reference to X_c and then augmented before being fed into the CLIP encoder $\Phi_I(x)$ and the surrogate model $f_1(\cdot)$. Thus, we compute the embeddings for X_c and X_p based on $f_1(\cdot)$, respectively: $E_c \leftarrow \frac{1}{|X_c|} \sum f_1(X_c)$, $E_p \leftarrow f_1(X_p)$, where the embeddings are extracted from the second-to-last layer, E_c is the averaged embeddings. Then, for each $x_i \in X_p$, compute cosine similarity between its embedding and E_c:

$$F_{\text{filter}}(x_i) = \frac{E_p^i \cdot E_c}{\left\|E_p^i\right\| \cdot \|E_c\|} \tag{4}$$

where E_p^i is the embedding of x_i. The output of $F_{\text{filter}}(x_i)$ is further normalized to the range $(0, 1)$. x_i with similarity $\text{sim}(x_i, E_c)$ larger than σ are considered to have semantics of the target class, which are filtered. σ is determined based on the statistics of cosine similarity. Line 3 in Algorithm 1 indicates that only examples that lack confusing semantics for the target class can be further applied to optimize the adversarial perturbation.

We then apply random augmentations, including rotation, scaling, translation, and patching, to increase variations in the features of examples, as inspired by prior works (Xu et al. 2025). These augmentations are applied randomly with varying strength and parameters to enhance the diversity of examples, thereby reducing overfitting of semantics of AEs. As illustrated in Line 4-5 in Algorithm 1, the remained example x_p and perturbed example $x_p + \delta$ are all augmented.

Experiments

Experimental Setup

The setups of datasets, models, CIL methods and baselines are introduced, followed by evaluation metrics and baselines compared with SAE.

Datasets and Models. We consider two widely used benchmark datasets: CIFAR-100 (Krizhevsky, Hinton et al. 2009) and ImageNet-100 (Rebuffi et al. 2017b), each comprising 100 classes. Following recent CIL research (Zhou et al. 2024b), both datasets are partitioned into 10 groups of 10 classes to simulate the CIL process. The target class is selected from the first 10 classes. We use ResNet-32 for

[3] For notational simplicity, only here we use the notation of y_p, y_t and Y_{nt} to denote the texts of the corresponding classes.

Dataset	Attack	Finetune	Replay	MEMO	DER	Foster	WA	BiC	iCaRl	PodNet	AVG.
CIFAR-100	Clean Acc	26.23	58.38	71.69	70.90	64.89	65.54	60.39	62.85	55.65	59.61
	MIFGSM	7.29	13.41	21.71	36.85	17.93	16.51	18.95	17.94	44.17	21.64
	GAKer	0.04	0.02	0.01	0.00	0.00	0.04	0.00	0.00	5.43	0.62
	AIM	**9.94**	30.96	0.05	0.04	48.47	56.20	0.04	28.15	0.02	19.32
	CGNC	0.57	0.12	0.00	0.00	0.48	0.30	0.00	0.00	0.08	0.17
	CleanSheet	1.23	2.31	3.80	1.16	2.02	2.71	4.02	6.00	7.88	3.46
	UnivIntruder	4.85	31.23	34.82	40.23	31.23	41.82	33.62	32.43	42.20	32.49
	SAE (Ours)	**9.38**	**37.56**	**52.72**	**63.07**	**49.47**	**63.51**	**53.14**	**36.49**	**85.30**	**50.07**
ImageNet-100	Clean Acc	24.91	58.75	71.69	74.14	68.40	67.39	61.58	57.89	67.28	61.34
	MIFGSM	**9.80**	9.36	16.05	26.23	22.46	15.05	9.48	11.08	68.02	20.84
	GAKer	0.14	1.08	0.33	0.17	0.22	0.00	0.00	0.00	0.17	0.23
	AIM	0.23	4.46	2.26	4.34	1.77	1.09	1.60	1.60	3.60	2.33
	CGNC	0.07	0.11	0.05	0.06	0.05	0.05	0.06	0.01	0.06	0.06
	CleanSheet	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.47	0.00	0.05
	UnivIntruder	0.34	4.17	1.63	3.36	2.82	2.44	0.75	0.93	2.53	2.11
	SAE (Ours)	**6.04**	**18.92**	**32.88**	**32.62**	**22.53**	**24.09**	**24.70**	**21.74**	**83.89**	**29.71**

Table 1: Average SASR across ten target classes for various attacks and CIL methods. All CIL methods are trained using the CIFAR-100 and ImageNet-100 datasets. Clean Acc denotes the average accuracy of models trained across ten tasks.

CIFAR-100 and ResNet-50 for ImageNet-100, due to their effectiveness in image classification and suitability in CIL.

Specifically, we use Tiny-ImageNet (Le and Yang 2015), which contains 200 classes, as the POOD dataset for CIFAR-100. For ImageNet-100, we use ImageNet-1K, which includes 1000 classes. To ensure a clean separation, we exclude classes from ImageNet-1K that overlap with ImageNet-100, guaranteeing no overlap in classes or images between the POOD dataset and the CIL training set.

CIL Methods and Attack Baselines. We evaluate SAE on nine representative CIL methods, categorized into five paradigms: 1) Finetune, usually used as the baseline of CIL (Zhou et al. 2024b); 2) replay-based methods, including Replay (Ratcliff 1990); 3) dynamic network-based methods, such as MEMO (Zhou et al. 2022), DER (Buzzega et al. 2020), and Foster (Wang et al. 2022a); 4) model rectification-based methods, including WA (Zhao et al. 2020) and BiC (Wu et al. 2019); 5) knowledge distillation-based methods, represented by iCaRL (Rebuffi et al. 2017a) and PodNet (Douillard et al. 2020). These methods are implemented using a public CIL benchmark framework[4].

To ensure fair comparison, we select representative targeted transferable attacks from both generative and iterative paradigms as baselines. Specifically, we adopt AIM (Li, Ma, and Jiang 2025), GAKer (Sun et al. 2024), and CGNC (Fang et al. 2024) as generative baseline attacks. For iterative baseline attacks, we include MIFGSM (Dong et al. 2018), CleanSheet (Ge et al. 2024), and UnivIntruder (Xu et al. 2025).

Implementation Details and Metrics. We adopt CLIP as implemented in OpenCLIP (Radford et al. 2021). Specifically, we use the ViT-B-32 model pre-trained on the LAION-2B dataset (Schuhmann et al. 2022). For perturbation optimization, we employ the Adam optimizer (Diederik 2014) with a learning rate of 0.01, a weight decay of 1×10^{-5}, and a batch size of 256. The optimization is conducted for 50 epochs. Additionally, constraint for epsilon ϵ is set to $32/255$, following the standard setting for black-box transferable attacks (Chen et al. 2023; Xu et al. 2025). σ for filtering is set 0.7 and embeddings are extracted from the second-to-last layer of $f_1(\cdot)$. All experiments are conducted on an RTX 4060 GPU with 8GB of memory.

We evaluate performance using two metrics: Attack Success Rate (ASR) and Sustainable Attack Success Rate (SASR). ASR denotes the proportion of perturbed test examples that are classified into the target class by the model updated after the i-th task $f_i(\cdot)$. SASR evaluates the sustainability of adversarial examples across the entire CIL process, which is defined as:

$$\text{SASR} = \frac{1}{M \times C} \sum_{i=1}^{M} \sum_{j=1}^{C} \mathbb{I}[f_i(x_j + \delta) = y_t] \qquad (5)$$

where $\mathbb{I}(\cdot)$ is an indicator function, which is 1 when classified correctly and 0 otherwise, $x_j \in X_{test}$, $C = |X_{test}|$.

Results

Sustainability. We compare SAE with baseline attacks under various CIL methods. As illustrated in Table 1, attack performance is evaluated using the average SASR across ten target classes on CIFAR-100 and ImageNet-100 datasets. The best results are highlighted via underline formatting. SAE consistently outperforms competing approaches under most CIL methods, achieving substantial improvements. On CIFAR-100, our method surpasses the average of all baselines by 37.12% across all CIL methods. Similar trends are observed on the ImageNet-100 dataset, where SAE continues to outperform all baselines by 25.44% average. As a result, our method achieves an average performance improvement of 31.28% across both datasets. In this experiment, we observe variation in SASR across different CIL methods. Specifically, for AEs of SAE and some baselines such as UnivIntruder and AIM, their SASRs on Finetune are relatively low. We hypothesize that this variability is due

[4]https://github.com/LAMDA-CL/PyCIL

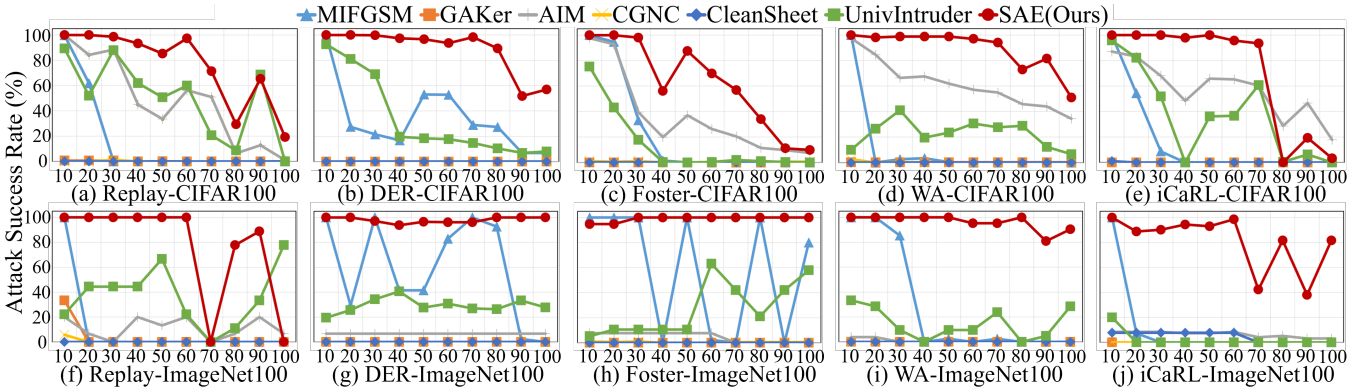

Figure 3: ASR curves for both baseline attacks and our attack across various CIL methods. Each subfigure illustrates the ASR across incremental tasks. Subfigures (a)–(e) present results for the CIFAR-100 dataset with 'skyscraper' as the target class. Subfigures (f)–(j) show the corresponding results for the ImageNet-100 dataset using 'candy store' as the target class.

CIL Method	'road'	'palm_tree'	'snake'	'bicycle'	'cloud'	'table'	'train'	'rabbit'	'shrew'	'skyscraper'
Replay	27.17	36.58	41.17	80.89	27.82	24.18	36.24	15.40	10.11	76.00
DER	62.92	96.53	24.96	99.57	5.41	88.79	94.18	19.70	50.19	88.39
Foster	23.04	80.13	14.04	72.64	17.63	48.30	99.82	46.96	29.84	62.35
WA	43.67	99.46	79.73	99.98	36.76	67.27	57.58	35.51	26.02	89.16
iCaRL	28.77	13.52	36.10	82.94	3.47	31.68	61.48	22.57	13.45	70.97
AVG.	**37.12**	**65.24**	**39.20**	**87.20**	**18.22**	**52.04**	**69.86**	**28.03**	**25.92**	**77.37**

Table 2: SASR of SAE across different ten attack target classes and five CIL methods, evaluated using models trained on the CIFAR-100 dataset. The results demonstrate the effectiveness of SAE in maintaining high ASR across diverse target classes.

to relatively stronger catastrophic forgetting in such a CIL method, which applies no strategies to mitigate the forgetting. In such a case, the CIL model struggles to correctly classify most samples (Clean Acc 26.23%), thus rendering the performance of AEs irrelevant.

We further analyze the performance of generative and iterative AEs. Among generative attacks, including GAKer, AIM, and CGNC, AIM demonstrates better sustainability than GAKer and CGNC due to its semantic injection module, which embeds target-class semantics into each layer of the generator for improved stability. For iterative attacks, i.e., MIFGSM, CleanSheet, and SAE, the results show that SAE significantly outperforms both MIFGSM and CleanSheet. MIFGSM relies heavily on gradients of $f_1(\cdot)$, making it prone to overfitting. On the other hand, CleanSheet focuses on universal target-class semantics but neglects the CIL model. Overall, SAE outperforms all the baselines, highlighting the effectiveness of our Semantic Correction Module, which prevents overfitting while ensuring the robustness of adversarial examples.

We also evaluate the performance of different adversarial attacks using the ASR across various task updates in the CIL model, as shown in Figure 3. The x-axis represents the number of learned classes, while the y-axis shows the ASR for each attack. Each subplot corresponds to a CIL method. From the results, ASR curves for SAE are consistently higher than baselines and exhibit significantly less fluctuation, indicating that the target-class semantics in SAE are more stable. When excluding the initial ASR in SASR, SAE achieves 35.45% avg. SASR (+28.76% vs. all baselines). Thus, SAE demonstrates greater robustness against CIL. It can also be observed that in CIL methods such as Replay and iCaRL, SAE experiences fluctuation in ASR when the number of learned classes exceeds about 70. This decline occurs because CIL methods rely on fixed-length caches to store previous data or model parameters to mitigate catastrophic forgetting (Zhou et al. 2024b). Once the cache is full, the model's ability to retain earlier knowledge weakens, causing a significant drop in ASR. Due to space constraints, we refer the interested reader to the extended version of our paper (Liu et al. 2025b) for all results.

Robustness to Target Class. To assess the robustness of SAE across different target classes, we performed targeted attacks on a range of classes. As shown in Table 2, we report the SASR across various CIL methods trained on the CIFAR-100 dataset. The results demonstrate that SAE maintains strong robustness across diverse target classes, with the average SASR exceeding 50% for all targets. A closer analysis reveals that SAE achieves notably high performance on most target classes, such as an average SASR of 87.2% for 'bicycle' and 77.37% for 'skyscraper'. For a small subset of classes, such as 'shrew' and 'cloud', the method yields relatively lower SASR values due to the confusing semantics of these classes. This variation stems from CIL class degradation rather than the shortage of our attack.

Perturbation's Visibility. In this paper, we constrain adversarial perturbations using the l_∞-norm. To explore

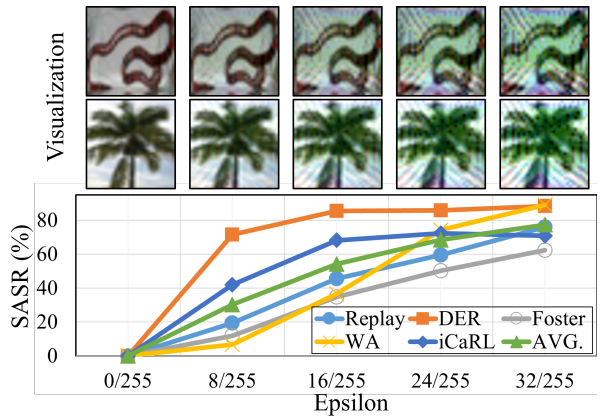

Figure 4: Perturbation's constraints and SASR on CIFAR-100. The target class is 'skyscraper'.

Defense	Foster		iCaRL	
	Clean	SASR	Clean	SASR
without Defense	71.15	62.35	75.28	70.97
Adversarial Training	58.30	51.26	61.57	41.26
Input Transformations	32.78	38.45	37.95	52.69
Feature Denoising	43.10	58.52	52.33	64.80
Random Smoothing	31.25	42.74	47.05	60.68

Table 3: Evaluation of four adversarial defenses on CIFAR-100 under clean and perturbed inputs. The target class is 'skyscraper'. The results also compare the Clean Accuracy of CIL and attack performance without and with defenses.

the impact of different perturbation constraints on performance, we conduct experiments under constraints of $8/255$, $16/255$, $24/255$, and $32/255$. The results, including the SASR and images illustrating perturbation perceptibility, are shown in Figure 4. The SASR exhibits a minor drop from 77.37% to 68.48% as the perturbation bound decreases from $32/255$ to $24/255$. It retains a strong performance of 54.15% even when the l_∞-norm is further reduced to $16/255$, indicating that SAE can generate effective adversarial examples even under more restrictive and imperceptible perturbations. Similar to CleanSheet (Ge et al. 2024) and UniverIntruder (Xu et al. 2025), the adversarial examples generated by SAE under various constraints are visually indistinguishable from their natural counterparts and remain imperceptible to human observers.

Defenses Evasion. In Table 3, we select four typical defense methods that are primarily used in CIL to enhance adversarial robustness (Cho, Lee, and Kim 2025) to evaluate SAE, including Adversarial Training (Liu et al. 2025a), Image Transformations (Meng and Chen 2017), Feature Denoising (Xie et al. 2019), and Random Smoothing (Jeong, Kim, and Shin 2023). The results show that SAE maintains an average SASR of 51.30%. Although these defenses enhance the robustness of models against SAE, they introduce a trade-off between model robustness and performance. As a result, these defenses may only be suitable for scenarios where high model accuracy is not a critical requirement.

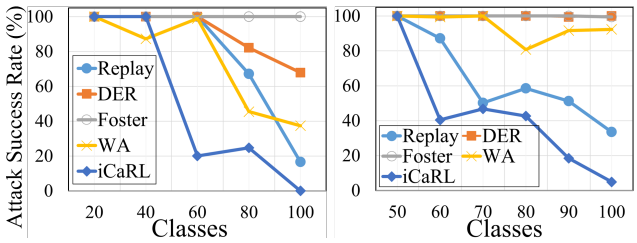

Figure 5: ASR curves for different CIL settings on CIFAR-100. The left plot illustrates ASR across five tasks, with CIL learning 20 new classes per task. The right plot illustrates ASR across six tasks, where CIL initially learns 50 classes, followed by incremental learning of 10 new classes per task.

Strategy	SASR	Strategy	SASR
No Module	21.64	FAM Only ($\sigma = 0.7$)	23.13
SCM Only	42.82	SCM & FAM ($\sigma = 0.5$)	48.91
FAM Only ($\sigma = 0.5$)	22.37	SCM & FAM ($\sigma = 0.7$)	50.07

Table 4: Ablation study of proposed modules. No Module indicates that only the gradients of the initial CIL model are used to optimize the adversarial samples. We also report the affect of various σ in FAM.

Ablation Study. We examine the impact of different CIL configurations on the SASR. As shown in Figure 5, our method maintains a consistent average SASR of 76.89% across different tasks, demonstrating the robustness of our approach and its ability to maintain high performance while adapting to new classes in diverse CIL scenarios.

Under the default attack evaluation setting, we evaluate the effectiveness of the two proposed modules, as shown in Table 4. SCM and FAM denote the Semantic Correction Module and the Filtering-and-Augmentation Module, respectively. From the results, applying the Semantic Correction Module increases the SASR by 21.18%. Adding the FAM further boosts SAE, outperforming the strategy without any modules by 28.43%. We further evaluate with $\sigma = 0.5$, which filters out more examples in FAM. Compared to the default setting of $\sigma = 0.7$, we observe a slight decline in performance, as many samples without confusing semantics are filtered out, resulting in fewer examples available for optimization.

Conclusion

In this paper, we propose SAE to enhance the sustainability of AEs in CIL. By addressing key challenges including overfitting and fluctuating AE semantics, we introduce two modules: the Semantic Correction Module, which uses a generative model and CIL gradients to ensure non-overfitted target-class semantics, and the Filtering-and-Augmentation Module, which eliminates examples with confusing semantics and augments the remained ones. Comprehensive experiments validate the effectiveness of SAE, demonstrating improved sustainability across various CIL.

Acknowledgments

We sincerely appreciate the anonymous reviewers for their insightful comments. This work was supported by the National Natural Science Foundation of China (U21A20464, U23A20306, U23A20307, U2436206, 62406239), the China Postdoctoral Science Foundation (No. 2023M742739), the '111 Center' (B16037), and the Fundamental Research Funds for the Central Universities (Program No.QTZX24081).

References

Ashtekar, N.; Zhu, J.; and Honavar, V. G. 2025. Class Incremental Learning from First Principles: A Review. *Transactions on Machine Learning Research*. Survey Certification.

Badjie, B.; Cecilio, J.; and Casimiro, A. 2024. Adversarial attacks and countermeasures on image classification-based deep learning models in autonomous driving systems: A systematic review. *ACM Computing Surveys*, 57(1): 1–52.

Buzzega, P.; Boschini, M.; Porrello, A.; Abati, D.; and Calderara, S. 2020. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33: 15920–15930.

Byun, J.; Kwon, M.-J.; Cho, S.; Kim, Y.; and Kim, C. 2023. Introducing competition to boost the transferability of targeted adversarial examples through clean feature mixup. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 24648–24657.

Chen, P.; Yang, J.; Lin, J.; Lu, Z.; Duan, Q.; and Chai, H. 2023. A practical clean-label backdoor attack with limited information in vertical federated learning. In *2023 IEEE International Conference on Data Mining (ICDM)*, 41–50. IEEE.

Chen, Z.; and Liu, B. 2018. *Lifelong machine learning*. Morgan & Claypool Publishers.

Cho, S.; Lee, H.; and Kim, C. 2025. Enhancing Robustness in Incremental Learning with Adversarial Training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2518–2526.

De Lange, M.; Aljundi, R.; Masana, M.; Parisot, S.; Jia, X.; Leonardis, A.; Slabaugh, G.; and Tuytelaars, T. 2022. A Continual Learning Survey: Defying Forgetting in Classification Tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7): 3366–3385.

Diederik, K. 2014. Adam: A method for stochastic optimization. *(No Title)*.

Dong, Y.; Liao, F.; Pang, T.; Su, H.; Zhu, J.; Hu, X.; and Li, J. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 9185–9193.

Douillard, A.; Cord, M.; Ollion, C.; Robert, T.; and Valle, E. 2020. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part XX 16*, 86–102. Springer.

Fang, H.; Kong, J.; Chen, B.; Dai, T.; Wu, H.; and Xia, S.-T. 2024. Clip-guided generative networks for transferable targeted adversarial attacks. In *European Conference on Computer Vision*, 1–19. Springer.

Gao, Z.; Han, S.; Zhang, X.; Xu, K.; Zhou, D.; Mao, X.; Dou, Y.; and Wang, H. 2025a. Maintaining Fairness in Logit-based Knowledge Distillation for Class-Incremental Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39: 16763–16771.

Gao, Z.; Jia, W.; Zhang, X.; Zhou, D.; Xu, K.; Dawei, F.; Dou, Y.; Mao, X.; and Wang, H. 2025b. Knowledge memorization and rumination for pre-trained model-based class-incremental learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 20523–20533.

Ge, Y.; Wang, Q.; Huang, H.; Li, Q.; Wang, C.; Shen, C.; Zhao, L.; Jiang, P.; Fang, Z.; and Zhang, S. 2024. Hijacking attacks against neural network by analyzing training data. In *33rd USENIX Security Symposium (USENIX Security 24)*, 6867–6884.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Guo, Q.; Pang, S.; Jia, X.; Liu, Y.; and Guo, Q. 2024. Efficient generation of targeted and transferable adversarial examples for vision-language models via diffusion models. *IEEE Transactions on Information Forensics and Security*.

Jeong, J.; Kim, S.; and Shin, J. 2023. Confidence-aware training of smoothed classifiers for certified robustness. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 8005–8013.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.(2009).

Le, Y.; and Yang, X. 2015. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7): 3.

Li, L.; Tan, Y.; Yang, S.; Cheng, H.; Dong, Y.; and Yang, L. 2025. Adaptive Decision Boundary for Few-Shot Class-Incremental Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 17, 18359–18367.

Li, T.; Ma, X.; and Jiang, Y.-G. 2025. AIM: Additional Image Guided Generation of Transferable Adversarial Attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 5, 4941–4949.

Liu, C.; Dong, Y.; Xiang, W.; Yang, X.; Su, H.; Zhu, J.; Chen, Y.; He, Y.; Xue, H.; and Zheng, S. 2025a. A comprehensive study on robustness of image classification models: Benchmarking and rethinking. *International Journal of Computer Vision*, 133(2): 567–589.

Liu, T.; Liu, X.; Dong, L.; Liu, Y.; Yang, Y.; and Ma, Z. 2025b. Improving Sustainability of Adversarial Examples in Class-Incremental Learning. arXiv:2511.09088.

Liu, Y.; Schiele, B.; and Sun, Q. 2021. RMM: reinforced memory management for class-incremental learning. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713845393.

Masana, M.; Liu, X.; Twardowski, B.; Menta, M.; Bagdanov, A. D.; and Van De Weijer, J. 2022. Class-incremental learning: survey and performance evaluation on image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5): 5513–5533.

Meng, D.; and Chen, H. 2017. Magnet: a two-pronged defense against adversarial examples. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, 135–147.

Naseer, M.; Khan, S.; Hayat, M.; Khan, F. S.; and Porikli, F. 2021. On generating transferable targeted perturbations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7708–7717.

Pelekis, S.; Koutroubas, T.; Blika, A.; Berdelis, A.; Karakolis, E.; Ntanos, C.; Spiliotis, E.; and Askounis, D. 2025. Adversarial machine learning: a review of methods, tools, and critical industry sectors. *Artificial Intelligence Review*, 58(8): 226.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PmLR.

Ratcliff, R. 1990. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, 97(2): 285.

Rebuffi, S.-A.; Kolesnikov, A.; Sperl, G.; and Lampert, C. H. 2017a. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2001–2010.

Rebuffi, S.-A.; Kolesnikov, A.; Sperl, G.; and Lampert, C. H. 2017b. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2001–2010.

Schuhmann, C.; Beaumont, R.; Vencu, R.; Gordon, C.; Wightman, R.; Cherti, M.; Coombes, T.; Katta, A.; Mullis, C.; Wortsman, M.; et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35: 25278–25294.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 618–626.

Sun, X.; Cheng, G.; Li, H.; Pei, L.; and Han, J. 2023. On single-model transferable targeted attacks: A closer look at decision-level optimization. *IEEE Transactions on Image Processing*, 32: 2972–2984.

Sun, Y.; Yuan, S.; Wang, X.; Gao, L.; and Song, J. 2024. Any target can be offense: Adversarial example generation via generalized latent infection. In *European Conference on Computer Vision*, 383–398. Springer.

Wang, F.-Y.; Zhou, D.-W.; Ye, H.-J.; and Zhan, D.-C. 2022a. Foster: Feature boosting and compression for class-incremental learning. In *European conference on computer vision*, 398–414. Springer.

Wang, L.; Zhang, X.; Su, H.; and Zhu, J. 2024. A Comprehensive Survey of Continual Learning: Theory, Method and Application. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. IEEE.

Wang, Z.; Yang, H.; Feng, Y.; Sun, P.; Guo, H.; Zhang, Z.; and Ren, K. 2023. Towards transferable targeted adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 20534–20543.

Wang, Z.; Zhang, Z.; Lee, C.-Y.; Zhang, H.; Sun, R.; Ren, X.; Su, G.; Perot, V.; Dy, J.; and Pfister, T. 2022b. Learning to Prompt for Continual Learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 139–149.

Weng, J.; Luo, Z.; and Li, S. 2025. Improving Transferable Targeted Adversarial Attack via Normalized Logit Calibration and Truncated Feature Mixing. *IEEE Transactions on Information Forensics and Security*.

Wu, Y.; Chen, Y.; Wang, L.; Ye, Y.; Liu, Z.; Guo, Y.; and Fu, Y. 2019. Large scale incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 374–382.

Xie, C.; Wu, Y.; Maaten, L. v. d.; Yuille, A. L.; and He, K. 2019. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 501–509.

Xu, B.; Dai, X.; Tang, D.; and Zhang, K. 2025. One Surrogate to Fool Them All: Universal, Transferable, and Targeted Adversarial Attacks with CLIP. In *Proceedings of the 2025 on ACM SIGSAC Conference on Computer and Communications Security*.

Zhang, J.; Liu, L.; Silven, O.; Pietikäinen, M.; and Hu, D. 2025. Few-shot class-incremental learning for classification and object detection: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhao, B.; Xiao, X.; Gan, G.; Zhang, B.; and Xia, S.-T. 2020. Maintaining discrimination and fairness in class incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13208–13217.

Zheng, B.; Zhou, D.-W.; Ye, H.-J.; and Zhan, D.-C. 2025. Task-Agnostic Guided Feature Expansion for Class-Incremental Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10099–10109.

Zhou, D.-W.; Sun, H.-L.; Ning, J.; Ye, H.-J.; and Zhan, D.-C. 2024a. Continual learning with pre-trained models: A survey. *arXiv preprint arXiv:2401.16386*.

Zhou, D.-W.; Wang, Q.-W.; Qi, Z.-H.; Ye, H.-J.; Zhan, D.-C.; and Liu, Z. 2024b. Class-incremental learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhou, D.-W.; Wang, Q.-W.; Qi, Z.-H.; Ye, H.-J.; Zhan, D.-C.; and Liu, Z. 2024c. Class-incremental learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhou, D.-W.; Wang, Q.-W.; Ye, H.-J.; and Zhan, D.-C. 2022. A model or 603 exemplars: Towards memory-efficient class-incremental learning. *arXiv preprint arXiv:2205.13218*.