

DeCLIP: Decoupled Prompting for CLIP-based Multi-Label Class-Incremental Learning

Kaile Du¹, Zihan Ye², Junzhou Xie¹, Yixi Shen³, Yuyang Li¹, Fuyuan Hu³,
Ling Shao², Guangcan Liu^{1*}, Joost van de Weijer⁴, and Fan Lyu^{4*}

¹ School of Automation, Southeast University, China

² University of the Chinese Academy of Sciences, China

³ Suzhou University of Science and Technology, China

⁴ Computer Vision Center, Spain

Abstract. Multi-label class-incremental learning (MLCIL) continuously expands the label space while recognizing multiple co-occurring classes, making it prone to catastrophic forgetting and high false-positive rates (FPR). Extending CLIP to MLCIL is non-trivial because co-occurring categories violate CLIP’s single image–text alignment paradigm and task-level partial labeling induces high FPR. We propose DeCLIP, a replay-free and parameter-efficient framework that decouples CLIP representations via a one-to-one class-specific prompting scheme. By assigning each category its own prompt space, DeCLIP prevents semantic confusion across labels and decouples multi-label images into per-class views compatible with CLIP pre-training. The learned prompts are preserved as knowledge anchors, mitigating catastrophic forgetting without replay. We further introduce Adaptive Similarity Tempering (AST), a task-aware strategy that suppresses FPR without dataset-specific tuning. Experiments on MS-COCO and PASCAL VOC show that DeCLIP consistently outperforms prior methods with minimal trainable parameters.

Keywords: multi-label class-incremental learning, CLIP, false-positive suppression, replay-free

1 Introduction

Multi-label class-incremental learning (MLCIL) [5, 9] is a challenging paradigm in which a model must learn to recognize multiple co-occurring classes while the label space continuously expands over time, making it susceptible to catastrophic forgetting [27]. Unlike single-label class-incremental learning (SLCIL) [1, 14, 31], each image in MLCIL may contain multiple classes [12, 32], better reflecting real-world complexity but also substantially increasing the difficulty of continual adaptation. MLCIL typically adopts a task-level partial labeling scheme [10, 40], where only the labels of the current task are available, while annotations for past and future classes appearing in the same images remain unobserved.

Correspondence to: Guangcan Liu and Fan Lyu.

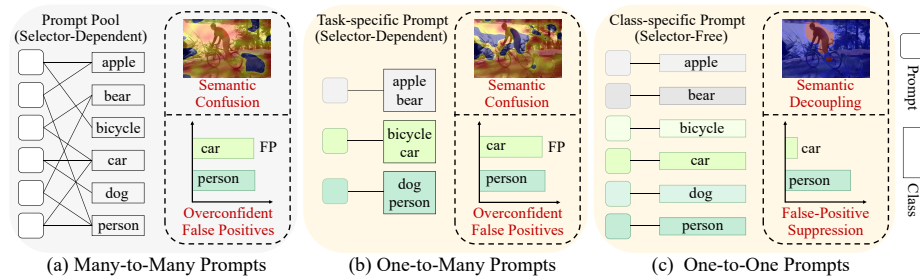


Fig. 1: Comparison of prompt-based methods. (a) Many-to-many prompts (L2P [37], DualPrompt [36]) couple co-occurring classes, causing semantic confusion and overconfident false positives (FP). (b) One-to-many prompts (MULTI-LANE [4]) still share prompt space across co-occurring classes. (c) DeCLIP uses one-to-one class-specific prompts for semantic decoupling and AST for false-positive suppression.

With the rapid development of large-scale pre-trained models, such models have demonstrated strong stability and generalization capabilities for CIL [14, 31, 36, 37]. Among them, vision–language pre-trained models such as CLIP [25] exhibit remarkable transferability to downstream tasks. As a result, CLIP-based CIL has recently emerged as a promising research direction [13–15, 34, 39]. SLCIL is well aligned with CLIP’s single-label pre-training paradigm, enabling strong performance even without any training, as shown by CL-CLIP [34].

In contrast, extending CLIP to MLCIL is non-trivial. In MLCIL, each image may contain multiple co-occurring classes [28] under a task-level partial labeling scheme, which fundamentally deviates from CLIP’s single-label alignment. As a result, naive CLIP-based approach CL-CLIP [34] that performs well in SLCIL degrades severely in MLCIL. This gap poses significant challenges for effectively leveraging CLIP in MLCIL, while simultaneously mitigating catastrophic forgetting. To address these challenges, we adopt a prompt-based strategy that enables parameter-efficient continual adaptation.

Prompt-based methods such as L2P [37] and DualPrompt [36] select prompts from a shared pool, forming a **many-to-many** prompt-class mapping, where a prompt may guide multiple classes and a class may be influenced by multiple prompts (Figure 1 (a)). MULTI-LANE [4] shares a single prompt across all classes within each task, resulting in a **one-to-many** mapping (Figure 1 (b)). These class-agnostic prompt mappings are misaligned with CLIP’s training paradigm based on single image-text pairs, as they let co-occurring categories share prompt space and thus entangle semantics, causing **semantic confusion** and blurring class boundaries. For example, in Figure 1 (b), the same prompt is shared by both “person” and “dog”; activating one class also triggers features of the other, blurring class boundaries and degrading per-class discriminability. Furthermore, relying on the prompt pool and selector inevitably perturbs prompts learned for earlier tasks, which exacerbates catastrophic forgetting [16, 44]. Moreover, under task-level partial labeling, many negative labels from past and future

are missing at each step [9, 10]. As tasks accumulate, negative evidence becomes systematically under-trained and the model tends to assign spuriously high confidence to absent classes (e.g., “car”), leading to many **overconfident false positives** shown in Figure 1. Therefore, effectively adapting CLIP to MLCIL calls for a dedicated strategy to suppress false-positive rates (FPR).

DeCLIP is proposed to address the above issues, as shown in Figure 1 (c). For **semantic decoupling**, DeCLIP adopts a **one-to-one** class-to-prompt design with *class-specific prompting* in both visual and textual modalities, where each class owns a dedicated prompt space. Specifically, for each class, we learn a highly lightweight prompt that guides the frozen CLIP visual encoder to extract a class-specific view of the image, which is then matched against the corresponding text feature. Consequently, a multi-label image is decomposed into class-specific views that better match CLIP’s single image–text alignment. The learned class-specific prompts are highly discriminative and are preserved as knowledge-preserving anchors for past classes. The selector-free design keeps these class-specific anchors from being interfered with by later tasks, thereby more effectively mitigating catastrophic forgetting without any replay. For **false-positive suppression**, we introduce *adaptive similarity tempering* (AST) to handle the intrinsically high FPR induced by task-level partial labeling. AST modulates each class-specific similarity pair in the vision–language space using a task-aware temperature schedule at inference, suppressing overconfident predictions without dataset- or scenario-specific hyperparameter tuning. We further adopt late-layer prompting, optimizing the position and manner of prompt attachment within the visual encoder. Our contributions are summarized as follows:

- (1) We propose DeCLIP, the first replay-free CLIP-based MLCIL framework that performs semantic decoupling via one-to-one class-specific prompting, decomposing each multi-label image into class-wise views and preserving learned prompts as knowledge anchors to mitigate catastrophic forgetting.
- (2) We introduce a task-aware adaptive similarity tempering strategy that modulates per-class vision-language similarities at inference for false-positive suppression without dataset- or scenario-specific tuning.
- (3) Extensive experiments demonstrate that our method consistently outperforms recent MLCIL methods with limited trainable parameters.

2 Related Work

Single-label class-incremental learning (SLCIL). Recent years have witnessed substantial progress in SLCIL. Regularization-based approaches [19, 21, 24, 43] alleviate forgetting by adding penalty terms to the loss. Replay-based methods [2, 27, 30] mitigate forgetting by maintaining a memory. Prompt-based methods leverage pre-trained models to enhance performance. L2P [37] and DualPrompt [36] learn a prompt pool and select them for different inputs through a many-to-many mechanism. Recently, vision–language pre-trained model [25] exhibits remarkable performance. CLAP [15] is built upon a probabilistic modeling paradigm, while MOE4CL [39] adopts a mixture-of-experts architecture with a

routing mechanism to preserve knowledge, RAPF [13] inserts a linear layer after the visual encoder. MG-CLIP [14] is the state-of-the-art SLCIL method, which mitigates the modality gap and introduces a compensation mechanism.

Multi-label class-incremental learning (MLCIL). MLCIL has recently gained increasing attention. Replay-based approaches such as PRS [17] and OCDM [22] adopt tailored sampling to alleviate the long-tailed label distributions, while CUTER [35] introduces a cut-out and experience replay strategy. Regularization-based methods include KRT [5], which proposes a knowledge restoration and transfer framework. CSC [9] employs a learnable GCN to recalibrate label dependencies. HCP [40] reduces ambiguity between known and unknown knowledge. RebLL [10] tackles the intrinsic positive-negative imbalance via asymmetric loss. L3A [41] proposes label-augmented analytic adaptation for MLCIL, but adopts a different evaluation criterion. Meanwhile, prompt-based methods have also emerged as promising solutions. MULTI-LANE [4] introduces patch selectors and one-to-many task-specific prompts, whereas DPA [42] leverages vision-language model with experience replay to alleviate forgetting. We propose the first replay-free CLIP-based MLCIL framework.

3 Method

3.1 Problem Formulation and Overall Framework

Problem formulation. MLCIL is defined over a sequence of T tasks. Each task $t \in \{1, \dots, T\}$ is associated with a training set $\mathcal{D}_{\text{trn}}^t$ and a testing set $\mathcal{D}_{\text{tst}}^t$, with its own label space $\mathcal{C}^t$. The cumulative label space up to task t is denoted as $\mathcal{C}^{1:t} = \bigcup_{i=1}^t \mathcal{C}^i$, where the label sets are disjoint, $\mathcal{C}^i \cap \mathcal{C}^j = \emptyset$ for $i \neq j$. Under the task-level partial labeling setting, only the current label set $\mathcal{Y}_{\text{trn}}^t = \mathcal{C}^t$ is available during training. At inference step t , the model is required to predict over the expanded label space $\mathcal{Y}_{\text{tst}}^t = \mathcal{C}^{1:t}$, ensuring that after completing task t it can recognize all categories encountered so far.

Overall framework. We ground our method on a pre-trained CLIP model [25] with a frozen visual encoder E_V and text encoder E_T . Figure 2 overviews DeCLIP, consisting of two components: (1) *class-specific prompting* for **semantic decoupling**, which establishes a one-to-one correspondence between prompts and classes to decouple co-occurring categories into class-wise views and mitigate catastrophic forgetting in a replay-free manner; and (2) *adaptive similarity tempering* for **false-positive suppression**, which modulates per-class vision-language similarities to suppress high false-positive rates. We also design a tailored prompting optimization strategy. The details are as follows.

3.2 DeCLIP: Decoupled Prompting for CLIP-based MLCIL

We realize decoupled prompting via a one-to-one class-specific prompting scheme.

Class-specific prompting. Existing prompt-based CIL methods typically employ class-agnostic designs, realizing either many-to-many [36, 37] or one-to-many [4] prompt-class mappings. Such designs force co-occurring classes to share

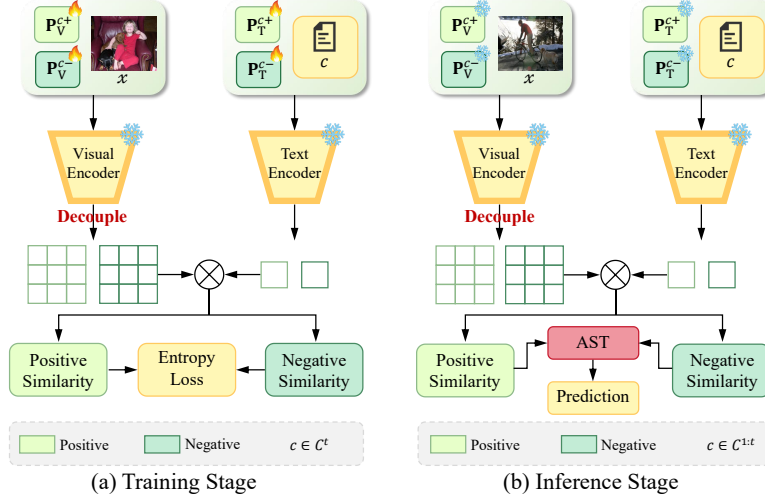


Fig. 2: The overall pipeline of DeCLIP. (a) Training stage on task t . For each class $c \in \mathcal{C}^t$, positive and negative prompts guide the frozen encoders to decouple features into class-specific components, from which positive and negative similarities are computed for optimization. (b) Inference stage after task t . All learned prompts corresponding to classes $c \in \mathcal{C}^{1:t}$ are fixed and jointly applied. The positive and negative similarities are adaptively calibrated through the AST module to produce the final prediction.

the same prompt space, leading to **semantic confusion**. Moreover, the selector-dependent mechanism inevitably disrupts the prompts learned for previous tasks, thereby aggravating catastrophic forgetting [16, 44]. To address these issues, we propose a class-specific and selector-free prompting strategy that enforces a one-to-one mapping for **semantic decoupling** and preserves class-specific prompts as lightweight knowledge anchors, thereby more effectively mitigating catastrophic forgetting with negligible parameter overhead.

Specifically, for each category c , we construct a lightweight one-to-one prompt set $\mathcal{P}^c = \{\mathbf{P}_T^c, \mathbf{P}_V^c\}$, where $\mathbf{P}_T^c = \{\mathbf{P}_T^{c+}, \mathbf{P}_T^{c-}\}$ are text prompts and $\mathbf{P}_V^c = \{\mathbf{P}_V^{c+}, \mathbf{P}_V^{c-}\}$ are visual prompts shown in Figure 2. Here, the “+” positive prompt encodes the presence of class c and boosts its similarity, whereas the “−” negative prompt encodes the absence of c . This one-to-one design explicitly decouples class representations by assigning each co-occurring category its own prompt space, preventing semantic confusion across labels.

This positive-negative design reformulates multi-label recognition as a set of binary classification tasks, ensuring that each category is modeled with both inclusion and exclusion cues. To align textual and visual features for binary classification, we compute two similarities as illustrated in Figure 2: a positive similarity s^{c+} that measures evidence for the presence of c , and a negative similarity s^{c-} that measures evidence for its absence. For each category $c \in \mathcal{C}^t$ in training stage (Figure 2 (a)) or $c \in \mathcal{C}^{1:t}$ in inference stage (Figure 2 (b)), text

features are obtained by combining the category name with the positive and negative text prompts in E_T , while class-decoupled visual features are obtained by injecting the class-specific prompts and image x into E_V . The two modalities are then aligned through cosine similarity:

$$\begin{aligned} s^{c+} &= \cos(E_T(\mathbf{P}_T^{c+}, c), E_V(\mathbf{P}_V^{c+}, x)), \\ s^{c-} &= \cos(E_T(\mathbf{P}_T^{c-}, c), E_V(\mathbf{P}_V^{c-}, x)). \end{aligned} \quad (1)$$

Here, $E_T(\cdot) \in \mathbb{R}^d$ outputs a single text embedding for class c , while $E_V(\cdot) \in \mathbb{R}^{L \times d}$ produces a set of L visual tokens for an input image x , where L denotes the number of visual tokens and d is the shared embedding dimension. Following common practice [12, 32, 42], cosine similarity is computed along the embedding dimension d between the text embedding and each visual token, yielding fine-grained token-wise similarity scores as illustrated in Figure 2. The token-wise similarities are then aggregated over the token dimension L to obtain a single scalar score $s^{c\pm}$ for each class.

The pair of similarities (s^{c+}, s^{c-}) is normalized by a binary softmax:

$$\hat{y}_{c+}^t = \frac{\exp(s^{c+}/\tau(t))}{\exp(s^{c+}/\tau(t)) + \exp(s^{c-}/\tau(t))}. \quad (2)$$

Here, $\hat{y}_{c+}^t$ denotes the confidence that class c is present in image x , while the negative prediction is defined as $\hat{y}_{c-}^t = 1 - \hat{y}_{c+}^t$. This similarity tempering mechanism sets $\tau(t) = 1$ during training and adjusts $\tau(t)$ in AST at inference to calibrate prediction confidence, which will be described next. Once a class $c \in \mathcal{C}^t$ has been learned through the training loss, its corresponding prompt is frozen and preserved as a lightweight knowledge anchor. As shown in Figure 2 (b), after training on task t , these preserved prompts $\{\mathbf{P}_V^c, \mathbf{P}_T^c\}, c \in \mathcal{C}^{1:t}$, are used during subsequent inference, providing stable class-specific references that help mitigate catastrophic forgetting without requiring any replay.

Adaptive similarity tempering. While semantic decoupling clarifies what to attend to, it does not determine how certain the model should be. In MLCIL, task-level partial labeling inevitably leads to many **overconfident false positives** [9, 10]. The past and future labels are missing at each step, and in practice most of these missing labels correspond to negatives. This causes negative evidence to be systematically under-trained, so the model tends to assign spuriously high confidence to many absent classes, making a high false-positive rate (FPR) issue in MLCIL. This phenomenon has already been verified in prior MLCIL works [9, 10] built on conventional backbones, and we observe the same behavior in the vision-language model. This effect is evident from the polarized confidence distribution in Figure 3 (a), where the model yields an FPR of 25.4%.

For **false-positive suppression**, we introduce an **adaptive similarity tempering** (AST) module, which modulates each positive-negative similarity pair at inference shown in Figure 2 (b). Existing approaches for mitigating FPR mainly rely on generic regularizers such as the maximum-entropy objective [9] or asymmetric loss (AL) [10] that aggressively down-weight positives. However,

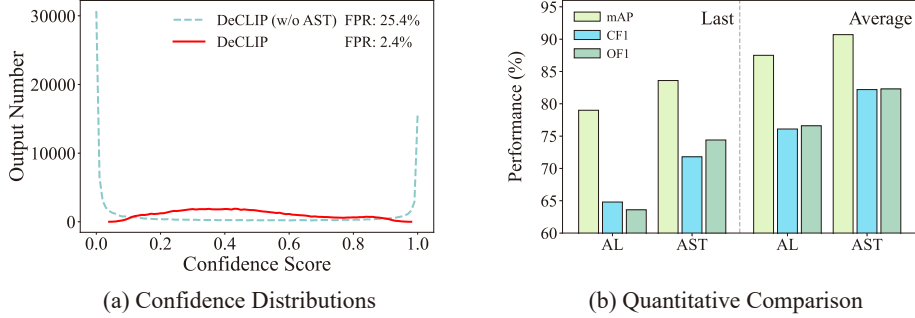


Fig. 3: (a) Confidence distributions with and without AST, showing that AST significantly suppresses false positives and reduces the FPR from 25.4% to 2.4% in VOC B4-C2. (b) Comparison between AL and AST, indicating that AST is tailored to CLIP-based MLCIL and yields better performance across both Last and Average evaluations.

when transplanted to CLIP-based MLCIL, these methods may be problematic for two reasons: (1) When operating on prompts, down-weighting positives can suppress informative positive evidence, thereby amplifying prompt forgetting and degrading the semantic anchors. (2) The strengths of these regularizers act as hyperparameters whose optimal values vary across datasets and class-incremental scenarios, requiring careful per-setting tuning.

By contrast, our AST operates at the level of each positive-negative similarity pair. We build on the two-way softmax in Eq. (2), where each pair (s^{c+}, s^{c-}) is normalized with a task-aware temperature. To realize a task-aware yet dataset-robust schedule, we define the temperature $\tau(t)$ as

$$\tau(t) = \begin{cases} 1, & t = 1, \\ \max\left(\lambda \frac{t}{|\mathcal{C}^{1:t}|}, 1\right), & \text{otherwise,} \end{cases} \quad (3)$$

where $|\mathcal{C}^{1:t}|$ denotes the number of classes learned up to task t , λ is a scalar constant shared across all datasets and incremental scenarios, following [3]. This schedule guarantees $\tau(t) \geq 1$ and gradually increases the amount of tempering as more tasks are learned, while automatically adapting to different class-incremental configurations through $|\mathcal{C}^{1:t}|$. Crucially, it is dataset- and scenario-robust, avoiding the per-dataset hyperparameter retuning required by previous regularization-based remedies [9, 10]. As shown by comparing Figure 3 (a), tempering the positive-negative similarity in this way drastically reduces the heavy tail of overconfident false positives (from 25.4% to 2.4%). Figure 3 (b) presents a quantitative comparison between AL [10] and our AST under both Last and Average metrics in VOC B4-C2. AST consistently outperforms AL in terms of mAP, CF1 and OF1, indicating more reliable multi-label predictions.

In summary, DeCLIP assigns each class a pair of positive and negative prompts, thereby decoupling co-occurring categories into class-wise views and

aligning vision-language similarities. AST then tempers the positive similarity in the CLIP-based binary softmax as tasks accumulate. Together, these two components close the loop from semantic understanding to confidence judgment in CLIP-based MLCIL. The pseudocode for both training and inference is provided in **supplementary material**. Next, we present our optimization strategy tailored to our proposed method.

3.3 Optimization

We introduce late-layer prompting for optimization:

(1) **Attachment position of prompts**: In contrast to prior prompt-based methods [4, 31], which insert prompts into shallow layers primarily for efficiency under a class-agnostic setting, we observe that deeper layers encode richer semantic information [26] and are better suited for class-specific DeCLIP. Empirically, prompts placed in deeper layers consistently yield stronger performance. Based on this observation, we insert prompts into the last five layers, which reduces the number of backpropagated layers compared to using all layers.

(2) **Attachment manner of prompts**: We adopt lightweight and layer-shared prompting. Specifically, we define the class-specific prompt embeddings $\mathbf{P}^c \in \mathbb{R}^{L_P \times d}$ where L_P is the prompt length. Let the multi-head self-attention (MSA) layer with input $\mathbf{h} \in \mathbb{R}^{L \times d}$. The query, key, and value inputs of the MSA layer are $\mathbf{h}_Q, \mathbf{h}_K, \mathbf{h}_V$. The MSA output is given by:

$$\begin{aligned} \text{MSA}(\mathbf{h}_Q, \mathbf{h}_K, \mathbf{h}_V) &= \text{Concat}(\mathbf{h}_1, \dots, \mathbf{h}_m) W^O, \\ \text{where } \mathbf{h}_i &= \text{Attention}\left(\mathbf{h}_Q W_i^Q, \mathbf{h}_K W_i^K, \mathbf{h}_V W_i^V\right), \end{aligned} \quad (4)$$

where W^O , W_i^Q , W_i^K , and W_i^V are projection matrices. m is the number of heads, $\mathbf{h}_Q = \mathbf{h}_K = \mathbf{h}_V$. Then, we integrate class-specific prompts into MSA:

$$f(\mathbf{P}^c, \mathbf{h}) = \text{MSA}([\mathbf{P}^c; \mathbf{h}_Q], [\mathbf{P}^c; \mathbf{h}_K], [\mathbf{P}^c; \mathbf{h}_V]). \quad (5)$$

Unlike conventional prompt-tuning [20], we slice the output of last five MSA layers as $f(\mathbf{P}^c, \mathbf{h})[L_P:] \in \mathbb{R}^{L \times d}$ to keep dimensional consistency, enabling subsequent layers to match the backbone while still leveraging prompt interactions. More details about prompting can be found in **supplementary material**.

Finally, we adopt a binary cross-entropy (BCE) objective:

$$L_{\text{bce}} = \sum_{c \in \mathcal{C}^t} -y_c^t \log(\hat{y}_{c+}^t) - (1 - y_c^t) \log(\hat{y}_{c-}^t), \quad (6)$$

where $y_c^t \in \{0, 1\}$ denotes the ground-truth of category c at task t . The terms $\hat{y}_{c+}^t$ and $\hat{y}_{c-}^t$ represent the predicted probabilities obtained from the positive and negative similarity scores.

Table 1: Multi-label class-incremental results on MS-COCO datasets (%). A memory size of 0 indicates replay-free. Type S and M denote SLCIL and MLCIL methods. Results are grouped by backbone families: TResNetM, ViT-B/16, and CLIP with ViT-B/16. All results of our method are averaged over three runs.

Method	Type	Memory	MS-COCO B40-C10				MS-COCO B0-C10				
			Avg.		Last		Avg.		Last		
			mAP	mAP	CF1	OF1	mAP	mAP	CF1	OF1	
TResNetM	TPCIL [33]	S	5/class	63.1	53.1	25.3	25.1	63.8	50.8	20.1	21.6
	PODNet [7]	S		65.4	57.8	24.2	23.4	65.7	53.4	13.6	17.3
	DER++ [1]	S		69.6	59.0	41.9	43.7	68.1	54.6	33.3	36.7
	iCaRL [27]	S	20/class	65.6	55.7	22.1	25.5	59.7	43.8	19.3	22.8
	BiC [38]	S		65.5	55.9	38.1	40.7	65.0	51.1	31.0	38.1
	ER [30]	S		68.9	61.6	58.6	61.1	60.3	47.2	40.6	43.6
	PODNet [7]	S		71.0	64.2	46.6	42.1	70.0	58.8	45.2	48.7
	DER++ [1]	S	1000	73.6	66.3	51.5	53.5	72.7	63.1	45.2	48.7
	PRS [17]	M		50.8	33.2	9.3	15.1	48.8	27.9	8.5	14.7
	OCDM [22]	M		51.3	34.0	9.5	15.5	49.5	28.5	8.6	14.9
	CUTER [35]	M		-	-	-	-	60.1	47.8	35.9	39.2
	AGCN [8]	M	0	73.9	69.1	58.7	59.9	72.4	61.4	53.9	56.6
	KRT [5]	M		77.8	74.0	64.4	63.4	74.6	65.9	55.6	56.5
	CSC [9]	M		78.2	75.0	65.7	67.0	78.0	72.8	64.9	66.8
	HCP [40]	M		78.9	75.3	64.9	68.6	77.9	71.2	60.4	65.3
	RebLL [10]	M		76.4	72.5	60.7	64.9	77.2	70.4	60.9	63.8
	L3A [41]	M		79.9	78.8	0.16	0.15	81.5	77.6	0.04	0.03
ViT	L2P [37]	S	0	73.7	71.1	62.8	62.3	73.4	68.0	58.8	56.0
	DualPrompt [36]	S		74.3	71.4	63.0	64.3	74.1	68.6	55.3	56.3
	CODA-P [31]	S		73.0	73.0	65.4	66.6	75.8	70.5	59.0	59.4
	MULTI-LANE [4]	M		78.9	76.6	65.9	66.5	79.4	74.3	65.7	64.8
CLIP	CL-CLIP [34]	S	0	5.2	4.4	6.6	7.0	6.8	4.4	6.6	7.0
	CLAP [15]	S		55.8	56.2	62.9	62.5	65.0	55.6	63.3	69.8
	RAPF [13]	S		70.8	64.1	58.0	59.7	69.3	59.3	54.4	54.0
	MOE4CL [39]	S		75.4	68.4	62.2	59.7	77.9	66.9	60.0	54.0
	MG-CLIP [14]	S		74.8	71.0	61.3	65.2	77.8	70.4	60.8	63.7
	DPA [42]	M		81.1	77.2	68.2	68.5	80.7	73.3	65.1	65.1
	DeCLIP	M		84.1	81.4	70.9	71.9	84.0	78.3	68.7	70.0

4 Experiment

4.1 Experimental Settings

Dataset and evaluation. Following the experimental settings in [5, 9], we evaluate our approach on MS-COCO 2014 [23] and PASCAL VOC 2007 [11]. MS-COCO contains 82,081 training and 40,504 validation images spanning 80 categories. VOC provides 20 categories with 5,011 training and 4,952 test images. We report mean average precision (mAP), class-wise F1 score (CF1), and overall F1 score (OF1). For each metric, we provide the last model performance (Last) and the average performance (Avg.) across all incremental tasks.

Experimental setup. Following prior MLCIL studies [5, 10, 40], we describe different MLCIL settings using the notation Bx-Cy, where “x” denotes the number of classes in the base task and “y” the number of new classes introduced

Table 2: Multi-label class-incremental results on PASCAL VOC datasets (%).

Method	Type	Memory	VOC B0-C4				VOC B10-C2			
			Avg.	Last			Avg.	Last		
			mAP	mAP	CF1	OF1	mAP	mAP	CF1	OF1
TResNetM	ER [30]	S	82.6	67.7	47.2	47.2	78.8	64.3	40.8	37.6
	PODNet [7]	S	84.1	72.3	52.0	50.3	80.2	66.2	54.3	40.8
	DER++ [1]	S	84.9	74.4	59.8	56.4	81.6	67.1	55.5	45.9
	CUTER [35]	M	82.1	67.9	51.4	60.0	-	-	-	-
	OCDM [22]	M	76.1	45.4	36.4	40.5	-	-	-	-
	KRT [5]	M	91.8	84.2	60.1	59.8	84.9	72.0	25.8	38.7
	CSC [9]	M	90.4	85.1	67.7	62.2	89.0	83.8	62.0	47.2
	HCP [40]	M	92.9	87.9	-	-	90.1	81.9	-	-
	RebLL [10]	M	91.4	85.1	72.6	75.8	88.8	80.9	64.1	66.1
ViT	DualPrompt [36]	S	89.9	83.4	60.6	58.9	88.5	83.0	57.5	52.2
	CODA-P [31]	S	90.3	84.0	65.5	62.1	89.1	83.4	59.8	54.0
	MULTI-LANE [4]	M	93.5	88.8	74.8	69.9	93.0	88.0	63.8	49.4
CLIP	CL-CLIP [34]	S	13.8	7.7	12.4	13.2	9.4	7.7	12.4	13.2
	CLAP [15]	S	77.9	68.3	75.6	72.8	66.6	60.3	68.6	62.5
	RAPF [13]	S	84.6	71.2	67.7	66.6	80.3	68.7	39.2	38.5
	MOE4CL [39]	S	86.4	83.1	75.5	70.6	84.0	81.0	57.6	55.1
	MG-CLIP [14]	S	92.5	85.4	72.7	64.4	90.8	86.4	52.8	53.5
	DPA [42]	M	93.5	88.1	-	-	88.5	79.0	-	-
	DeCLIP	M	95.1	90.7	81.7	82.7	93.5	88.7	74.8	73.7

per incremental task. On COCO, we consider two standard scenarios (B0-C10, B40-C10) and two more challenging ones (B0-C5, B20-C4). For VOC, we adopt B0-C4, B10-C2 as standard cases and B5-C3, B4-C2 as more difficult settings.

Implementation details. Our method is implemented using the CLIP ViT-B/16 backbone [25]. For comparison, we include baselines built on the strong TResNetM [29] (designed for multi-label classification), and methods based on ViT-B/16 [6] and CLIP ViT-B/16 (same as ours). The input image size is 224×224 . We use Adam [18] to optimize the network with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The length of the prompts is set to 4 and $\lambda = 16$. The incremental sequence follows the lexicographic order of class names, with the same threshold as in [5, 9].

4.2 Overall Performance

We conduct a unified comparison of SLCIL and MLCIL using three backbones. The TResNetM-based group consists of KRT [5], CSC [9], RebLL [10] and L3A [41]. The vision-only group includes L2P [37], DualPrompt [36], CODA-P [31], and MULTI-LANE [4]. The vision-language CLIP-based group comprises CL-CLIP [34], CLAP [15], RAPF [13], MOE4CL [39], MG-CLIP [14], and DPA [42]. Most of the compared results are obtained from references [5, 9, 42] and are reproduced using their publicly released implementations.

MS-COCO. As shown in Table 1, on MS-COCO, our method consistently surpasses the existing SLCIL and MLCIL methods with three backbones across

Table 3: Long-sequence MLCIL results on MS-COCO dataset (%).

Method	Type	Memory	MS-COCO B20-C4				MS-COCO B0-C5			
			Avg.		Last		Avg.		Last	
			mAP	mAP	CF1	OF1	mAP	mAP	CF1	OF1
ER [30]	S	5/class	53.0	41.9	32.9	29.8	54.6	40.1	32.9	32.3
PODNet [7]	S		67.7	58.4	44.0	39.1	67.2	58.2	45.1	40.8
DER++ [1]	S		65.5	57.3	41.4	35.5	68.2	57.9	43.6	39.2
AGCN [8]	M	0	65.7	55.6	44.2	39.6	64.4	53.0	43.2	41.1
KRT [5]	M		64.0	45.2	17.6	33.0	63.1	44.5	22.6	37.5
CSC [9]	M		72.6	65.5	48.4	44.2	75.0	67.3	50.5	47.2
RebLL [10]	M		69.2	60.1	51.3	49.2	71.7	63.5	53.5	51.9
MULTI-LANE [4]	M	0	76.3	71.0	50.8	40.6	76.3	70.5	52.5	43.3
CL-CLIP [34]	S		5.7	4.4	6.6	7.0	7.7	4.4	6.6	7.0
RAPF [13]	S		62.2	53.3	41.7	35.4	63.4	53.7	48.1	39.1
MG-CLIP [14]	S		70.6	65.8	46.9	42.6	69.4	59.2	22.3	22.0
DeCLIP	M		79.6	73.8	54.2	58.3	80.4	73.9	59.1	63.3

Table 4: Long-sequence MLCIL results on PASCAL VOC dataset (%).

Method	Type	Memory	VOC B4-C2				VOC B5-C3			
			Avg.		Last		Avg.		Last	
			mAP	mAP	CF1	OF1	mAP	mAP	CF1	OF1
ER [30]	S	2/class	69.9	47.1	34.7	33.1	78.3	62.8	50.6	47.8
PODNet [7]	S		71.1	60.4	45.3	38.5	81.4	70.3	47.7	43.3
DER++ [1]	S		77.0	61.6	33.4	29.9	78.0	68.1	53.3	51.4
AGCN [8]	M	0	71.2	50.2	35.5	33.5	83.1	71.6	55.2	51.1
KRT [5]	M		71.0	43.6	13.7	31.0	86.2	74.6	39.3	46.6
CSC [9]	M		83.3	74.1	52.4	42.9	88.0	82.1	62.5	56.6
RebLL [10]	M		84.6	73.1	57.2	60.0	87.7	79.4	64.6	66.9
MULTI-LANE [4]	M	0	89.8	82.1	43.2	36.1	92.1	87.1	66.9	56.8
CL-CLIP [34]	S		12.9	7.7	12.4	13.2	12.1	7.7	12.4	13.2
RAPF [13]	S		80.8	66.2	27.3	25.7	83.5	68.4	50.1	46.1
MG-CLIP [14]	S		85.2	77.9	24.6	25.1	90.7	82.0	35.2	27.2
DeCLIP	M		90.8	83.6	71.8	74.4	93.0	88.2	77.3	80.2

all evaluation metrics. In B40-C10, our DeCLIP achieves 84.1% Avg. mAP and 81.4% Last mAP, clearly improving over the prior CLIP-based methods (e.g., DPA with 81.1% Avg. mAP). Similar gains are observed in B0-C10, indicating that DeCLIP effectively suppresses catastrophic forgetting.

PASCAL VOC. Table 2 reports the results on VOC. DeCLIP also outperforms previous methods across metrics under both incremental protocols. In B0-C4, DeCLIP reaches 90.7% Last mAP, along with 81.7% CF1 and 82.7% OF1, exceeding all competitors. The superiority remains consistent in B10-C2, showing that our method maintains stable learning and effectively mitigates forgetting.

Long-sequence settings. Under the long-sequence class-incremental setting, our method achieves consistent improvements over all compared approaches on both MS-COCO and PASCAL VOC, as shown in Table 3 and Table 4. In MS-COCO B20-C4 and B0-C5, DeCLIP outperforms every SLCIL and MLCIL baseline across all evaluation metrics. It achieves an average mAP of 79.6% and a

Table 5: Ablation of **class-specific prompts** for decoupling and **AST** under the last and average metrics (%).

Method	Last			Avg.		
	mAP	CF1	OF1	mAP	CF1	OF1
CL-CLIP [34]	7.7	12.4	13.2	12.9	20.1	20.6
+ Text Prompt	71.1	40.3	31.7	81.7	60.2	50.8
++ Visual Prompt	83.6	43.3	37.6	90.8	62.8	56.8
+++ AST	83.6	71.8	74.4	90.8	82.2	82.3

Table 6: Ablation of **positive-negative prompts** under the last and average metrics (%).

Method	Last			Avg.		
	mAP	CF1	OF1	mAP	CF1	OF1
CL-CLIP [34]	7.7	12.4	13.2	12.9	20.1	20.6
Neg. Prompts	76.2	58.8	58.9	86.0	70.2	70.1
Pos. Prompts	79.7	61.8	68.2	88.5	77.4	79.4
Neg. + Pos.	83.6	71.8	74.4	90.8	82.2	82.3

last-task mAP of 73.8% in B20-C4, and 80.4% and 73.9% in B0-C5. A similar trend holds on PASCAL VOC. Overall, the results in Table 3 and Table 4 demonstrate that our approach maintains strong robustness even in long-task sequences, largely due to its ability to effectively suppress forgetting and preserve discriminative multi-label prompts across tasks. Our mAP gains are more pronounced on the more complex COCO. Benefiting from our FPR suppression, the improvements in CF1 and OF1 can be more pronounced.

Training cost analysis. As shown in Figure 4, we analyze the trade-off between trainable parameters and performance in COCO B40-C10. Compared with prior CLIP-based approaches such as MG-CLIP [14], RAPF [13], MOE4CL [39], and DPA [42] using 1600 exemplars. Our method achieves the superior average mAP while requiring limited trainable parameters.

4.3 Ablation Study and Analysis

Ablation of different components.

Table 5 presents the ablation study on class-specific prompts for decoupling and the AST module. Starting from the CL-CLIP baseline, adding text prompts (Pos.+Neg.) yields a clear performance gain, and introducing visual prompts brings a further boost, highlighting the necessity and effectiveness of decoupling representations. With the full version that includes AST, the model suppresses overconfident predictions and reduces the FPR from 25.4% to 2.4%, while simultaneously improving both Last and Avg. performance. mAP is ranking-based and threshold-independent. AST is primarily designed to suppress FPR, which directly improves threshold-based metrics such as CF1 and OF1. Table 6 further analyzes the effect of introducing positive and negative

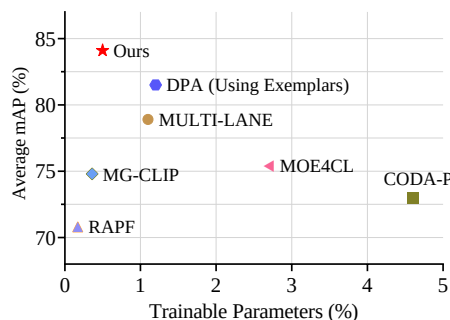
**Fig. 4:** Parameter-performance comparison of methods in MS-COCO B40-C10.

Table 7: Zero-shot cross-dataset transfer evaluation (%) on VOC, using the model incrementally trained on COCO under the B40-C10 and B0-C10 settings.

Method	Using B40-C10			Using B0-C10		
	mAP	CF1	OF1	mAP	CF1	OF1
CL-CLIP [34]	7.7	12.4	13.2	7.7	12.4	13.2
RAPF [13]	80.9	72.3	72.2	68.4	41.6	54.2
MG-CLIP [14]	85.2	35.7	47.2	63.8	43.6	48.8
DeCLIP	91.1	82.9	76.2	75.5	69.1	58.6

Table 8: Ablation study (%) of prompting strategies: One-to-All (one global prompt), One-to-Many (one prompt per task) and One-to-One (one prompt per class).

Method	Last			Avg.		
	mAP	CF1	OF1	mAP	CF1	OF1
CL-CLIP [34]	7.7	12.4	13.2	12.9	20.1	20.6
One-to-All	24.9	16.4	26.4	35.3	22.6	34.8
One-to-Many	32.6	24.2	31.4	41.5	34.8	44.2
One-to-One	83.6	71.8	74.4	90.8	82.2	82.3

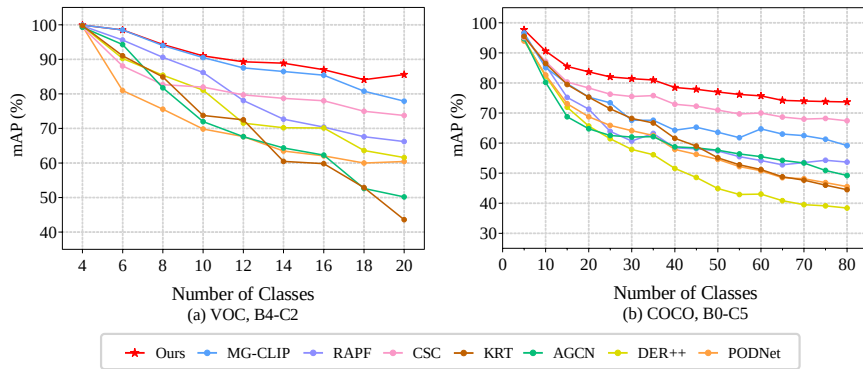


Fig. 5: The results of multi-task MLCIL on challenging VOC B4-C2 and COCO B0-C5, where a larger number of tasks are involved.

prompts across different modalities, which are used to train the model’s ability to distinguish the presence or absence of each class. Using either positive or negative prompts alone provides considerable benefits compared to the prompt-free CL-CLIP. Combining both leads to better performance across all metrics.

Zero-shot transfer ability. Following MG-CLIP [14], we also evaluate the zero-shot cross-dataset transfer ability on VOC (Last) using models that were incrementally trained on COCO. As shown in Table 7, we compare our method with CLIP-based methods that have comparable numbers of trainable parameters, namely RAPF and MG-CLIP. When using models trained under B40-C10 and B0-C10 settings, DeCLIP consistently outperforms these strong methods.

Ablation study of prompting strategies. Table 8 summarizes the ablation on different levels of prompting in VOC B4-C2. Compared with the prompt-free CL-CLIP, introducing a global prompt (one-to-all) improves the performance. Task-specific prompts (one-to-many) achieve higher results. Most notably, our class-specific prompts (one-to-one) deliver better performance, highlighting the importance of fine-grained semantic decoupling.

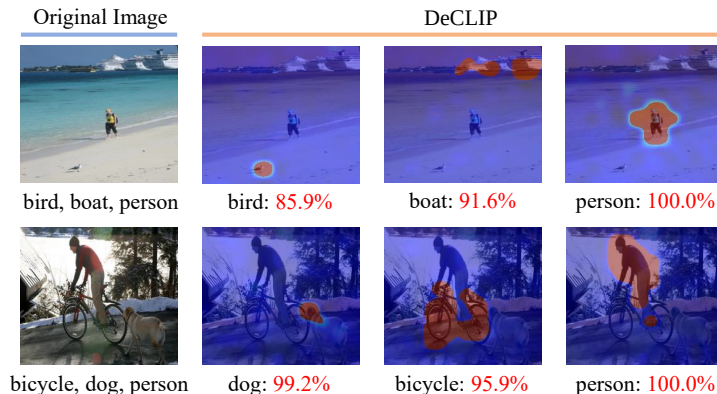


Fig. 6: Final-task model visualization across incremental stages.

Anti-forgetting performance. As shown in Figure 5, we evaluate the anti-forgetting performance of different methods in long-sequence VOC B4-C2 and COCO B0-C5. The results demonstrate that our approach achieves lower forgetting across tasks, indicating its stronger ability to preserve previously learned knowledge while adapting to new classes. Table 9 shows that our replay-free CLIP-based method outperforms some strong MLCIL approaches with replay, demonstrating that our design can effectively suppress catastrophic forgetting even without storing any memory samples.

Table 9: Comparison on mAP (%) between our replay-free method and MLCIL approaches using memory.

Method	Memory	B40-C10		B0-C10	
		Avg.	Last	Avg.	Last
AGCN-R [8]	20/class	75.2	71.7	73.2	66.0
KRT-R [5]		78.3	75.2	76.5	70.2
CSC-R [9]		78.7	76.7	79.6	74.8
DPA-R [42]		81.5	78.4	81.7	76.1
DeCLIP	0	84.1	81.4	84.0	78.3

Class activation maps. As shown in Figure 6, the original images are shown in the first column, followed by class activation maps (CAM) corresponding to each predicted category with confidence scores. For example, in the second row, the model correctly identifies “bicycle”, “dog” and “person”, and the highlighted regions precisely align with the respective objects. Notably, the body of “person”, including fine-grained parts such as the legs, can be clearly delineated, demonstrating the ability to capture detailed semantics. These results are obtained by testing the final-task model on classes across incremental stages, showing not only effective semantic decoupling but also strong anti-forgetting performance.

More ablation studies, t-SNE visualization, forgetting measure, sensitivity study, training and inference cost are provided in **supplementary material**.

5 Conclusion

MLCIL poses fundamental challenges to CLIP-based class-incremental learning due to semantic confusion among co-occurring categories and high false-positive rates caused by task-level partial labeling. To address these issues, we propose DeCLIP, a replay-free and parameter-efficient framework that decouples CLIP representations for MLCIL. DeCLIP performs class-specific prompting to achieve semantic decoupling, where the learned prompts are preserved as knowledge anchors to mitigate catastrophic forgetting without replay, while Adaptive Similarity Tempering (AST) modulates vision–language similarities to suppress false positives. In addition, late-layer prompting optimizes both the position and manner of prompt attachment, enabling efficient adaptation. As the first replay-free CLIP-based method for MLCIL, extensive experiments on MS-COCO and PASCAL VOC demonstrate that DeCLIP consistently outperforms prior approaches, validating its effectiveness.

Limitation. We adopt vanilla BCE to optimize DeCLIP for simplicity, leaving open the potential of tailored objectives to enhance cross-modal alignment.

References

1. Buzzega, P., Boschini, M., Porrello, A., Abati, D., Calderara, S.: Dark experience for general continual learning: a strong, simple baseline. In: *Proceedings of the Advances in Neural Information Processing Systems*. pp. 15920–15930 (2020)
2. Cha, S., Cho, S., Hwang, D., Hong, S., Lee, M., Moon, T.: Rebalancing batch normalization for exemplar-based class-incremental learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 20127–20136 (2023)
3. Cheng, J., Vasconcelos, N.: Towards calibrated multi-label deep neural networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 27589–27599 (2024)
4. De Min, T., Mancini, M., Lathuilière, S., Roy, S., Ricci, E.: Less is more: Summarizing patch tokens for efficient multi-label class-incremental learning. *arXiv preprint arXiv:2405.15633* (2024)
5. Dong, S., Luo, H., He, Y., Wei, X., Cheng, J., Gong, Y.: Knowledge restore and transfer for multi-label class-incremental learning. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 18711–18720 (2023)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: *Proceedings of the International Conference on Learning Representations* (2021)
7. Douillard, A., Cord, M., Ollion, C., Robert, T., Valle, E.: Podnet: Pooled outputs distillation for small-tasks incremental learning. In: *Proceedings of the European Conference on Computer Vision*. pp. 86–102 (2020)
8. Du, K., Lyu, F., Li, L., Hu, F., Feng, W., Xu, F., Xi, X., Cheng, H.: Multi-label continual learning using augmented graph convolutional network. *IEEE Transactions on Multimedia* **26**, 2978–2992 (2024)
9. Du, K., Zhou, Y., Lyu, F., Li, Y., Lu, C., Liu, G.: Confidence self-calibration for multi-label class-incremental learning. In: *European Conference on Computer Vision*. pp. 234–252 (2024)
10. Du, K., Zhou, Y., Lyu, F., Li, Y., Xie, J., Shen, Y., Hu, F., Liu, G.: Rebalancing multi-label class-incremental learning. In: *Proceedings of the AAAI conference on artificial intelligence*. vol. 39, pp. 16372–16380 (2025)
11. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. *International journal of computer vision* **88**(2), 303–338 (2010)
12. Hu, P., Sun, X., Sclaroff, S., Saenko, K.: Dualcoop++: Fast and effective adaptation to multi-label recognition with limited annotations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **46**(5), 3450–3462 (2023)
13. Huang, L., Cao, X., Lu, H., Liu, X.: Class-incremental learning with clip: Adaptive representation adjustment and parameter fusion. In: *European Conference on Computer Vision*. pp. 214–231. Springer (2024)
14. Huang, L., Cao, X., Lu, H., Meng, Y., Yang, F., Liu, X.: Mind the gap: Preserving and compensating for the modality gap in clip-based continual learning. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 3777–3786 (2025)
15. Jha, S., Gong, D., Yao, L.: Clap4clip: Continual learning with probabilistic finetuning for vision-language models. *Advances in neural information processing systems* **37**, 129146–129186 (2024)

16. Jung, D., Han, D., Bang, J., Song, H.: Generating instance-level prompts for rehearsal-free continual learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11847–11857 (2023)
17. Kim, C.D., Jeong, J., Kim, G.: Imbalanced continual learning with partitioning reservoir sampling. In: Proceedings of the European Conference on Computer Vision. pp. 411–428 (2020)
18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Proceedings of the International Conference on Learning Representations (2015)
19. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al.: Overcoming catastrophic forgetting in neural networks. *National Academy of Sciences* **114**(13), 3521–3526 (2017)
20. Lester, B., Al-Rfou, R., Constant, N.: The power of scale for parameter-efficient prompt tuning. arXiv preprint arXiv:2104.08691 (2021)
21. Li, Z., Hoiem, D.: Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40**(12), 2935–2947 (2017)
22. Liang, Y.S., Li, W.J.: Optimizing class distribution in memory for multi-label online continual learning. arXiv preprint arXiv:2209.11469 (2022)
23. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Proceedings of the European Conference on Computer Vision. pp. 740–755 (2014)
24. Mohamed, A., Grandhe, R., Joseph, K., Khan, S., Khan, F.: D3former: Debiased dual distilled transformer for incremental learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2420–2429 (2023)
25. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: Proceedings of the International Conference on Machine Learning. pp. 8748–8763 (2021)
26. Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., Dosovitskiy, A.: Do vision transformers see like convolutional neural networks? *Advances in neural information processing systems* **34**, 12116–12128 (2021)
27. Rebuffi, S.A., Kolesnikov, A., Sperl, G., Lampert, C.H.: icarl: Incremental classifier and representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2001–2010 (2017)
28. Ridnik, T., Ben-Baruch, E., Zamir, N., Noy, A., Friedman, I., Protter, M., Zelnik-Manor, L.: Asymmetric loss for multi-label classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 82–91 (2021)
29. Ridnik, T., Lawen, H., Noy, A., Ben Baruch, E., Sharir, G., Friedman, I.: Tresnet: High performance gpu-dedicated architecture. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1400–1409 (2021)
30. Rolnick, D., Ahuja, A., Schwarz, J., Lillicrap, T., Wayne, G.: Experience replay for continual learning. In: Proceedings of the Advances in Neural Information Processing Systems. pp. 350–360 (2019)
31. Smith, J.S., Karlinsky, L., Gutta, V., Cascante-Bonilla, P., Kim, D., Arbelle, A., Panda, R., Feris, R., Kira, Z.: 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)

32. Sun, X., Hu, P., Saenko, K.: Dualcoop: Fast adaptation to multi-label recognition with limited annotations. *Advances in Neural Information Processing Systems* **35**, 30569–30582 (2022)
33. Tao, X., Chang, X., Hong, X., Wei, X., Gong, Y.: Topology-preserving class-incremental learning. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX* 16. pp. 254–270. Springer (2020)
34. Thengane, V., Khan, S., Hayat, M., Khan, F.: Clip model is an efficient continual learner. *arXiv preprint arXiv:2210.03114* (2022)
35. Wang, X., Li, S.y., Zhang, J., Chen, S.: Cut out and replay: A simple yet versatile strategy for multi-label online continual learning. In: *Proceedings of the International Conference on Machine Learning* (2025)
36. Wang, Z., Zhang, Z., Ebrahimi, S., Sun, R., Zhang, H., Lee, C.Y., Ren, X., Su, G., Perot, V., Dy, J., et al.: Dualprompt: Complementary prompting for rehearsal-free continual learning. In: *European Conference on Computer Vision*. pp. 631–648. Springer (2022)
37. Wang, Z., Zhang, Z., Lee, C.Y., Zhang, H., Sun, R., Ren, X., Su, G., Perot, V., Dy, J., Pfister, T.: Learning to prompt for continual learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 139–149 (2022)
38. Wu, Y., Chen, Y., Wang, L., Ye, Y., Liu, Z., Guo, Y., Fu, Y.: Large scale incremental learning. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 374–382 (2019)
39. Yu, J., Zhuge, Y., Zhang, L., Hu, P., Wang, D., Lu, H., He, Y.: Boosting continual learning of vision-language models via mixture-of-experts adapters. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 23219–23230 (2024)
40. Zhang, A., Yang, D., Liu, C., Hong, X., Zhou, Y.: Specifying what you know or not for multi-label class-incremental learning. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 39, pp. 22345–22353 (2025)
41. Zhang, X., He, R., Jiao, C., Fang, D., Li, M., Zeng, Z., Chen, C., Zhuang, H.: L3a: Label-augmented analytic adaptation for multi-label class incremental learning. In: *Proceedings of the International Conference on Machine Learning*. pp. 74938–74949 (2025)
42. Zhao, H., Jin, Y., Ma, L.: Dynamic prompt adjustment for multi-label class-incremental learning. *arXiv preprint arXiv:2501.00340* (2024)
43. Zhao, L., Lu, J., Xu, Y., Cheng, Z., Guo, D., Niu, Y., Fang, X.: Few-shot class-incremental learning via class-aware bilateral distillation. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 11838–11847 (2023)
44. Zhou, D.W., Sun, H.L., Ye, H.J., Zhan, D.C.: Expandable subspace ensemble for pre-trained model-based class-incremental learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 23554–23564 (2024)