Generative Negative Text Replay for Continual Vision-Language Pretraining

Shipeng Yan[1,2,3], Lanqing Hong[4], Hang Xu[4], Jianhua Han[4], Tinne Tuytelaars[5], Zhenguo Li[4], and Xuming He[1,6]

[1] ShanghaiTech University [2] Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences [3] University of Chinese Academy of Sciences [4] Huawei Noah's Ark Lab [5] KU Leuven [6] Shanghai Engineering Research Center of Intelligent Vision and Imaging

yanshp@shanghaitech.edu.cn, {hanjianhua4, honglanqing, xu.hang, xuchunjing, li.zhenguo}@huawei.com, tinne.tuytelaars@kuleuven.be, hexm@shanghaitech.edu.cn

Abstract. Vision-language pre-training (VLP) has attracted increasing attention recently. With a large amount of image-text pairs, VLP models trained with contrastive loss have achieved impressive performance in various tasks, especially the zero-shot generalization on downstream datasets. In practical applications, however, massive data are usually collected in a streaming fashion, requiring VLP models to continuously integrate novel knowledge from incoming data and retain learned knowledge. In this work, we focus on learning a VLP model with sequential chunks of image-text pair data. To tackle the catastrophic forgetting issue in this multi-modal continual learning setting, we first introduce pseudo text replay that generates hard negative texts conditioned on the training images in memory, which not only better preserves learned knowledge but also improves the diversity of negative samples in the contrastive loss. Moreover, we propose multi-modal knowledge distillation between images and texts to align the instance-wise prediction between old and new models. We incrementally pre-train our model on the both instance and class incremental splits of Conceptual Caption dataset, and evaluate the model on zero-shot image classification and image-text retrieval tasks. Our method consistently outperforms the existing baselines with a large margin, which demonstrates its superiority. Notably, we realize an average performance boost of 4.60% on image-classification downstream datasets for class incremental split.

Keywords: Vision-Language Pretraining, Continual Learning

1 Introduction

Vision-and-language pre-training (VLP) [24,34] seeks to learn a generalizable multi-modal representations from large-scale image-text data. Recently, VLP models, such as CLIP [34], have demonstrated promising performance especially on the zero-shot generalization for a variety of downstream vision-language tasks

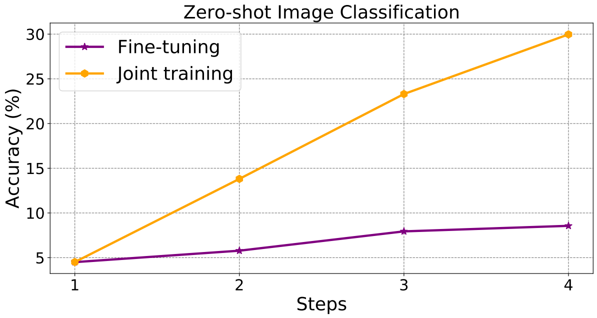

Fig. 1. Illustration of continual vision-and-language pre-training on CC2M dataset. The dataset is split into four data chunks with 0.5 million image-text pairs in each chunk. In fine-tuning, a new data chunk arrives at each step to update the pre-training model without previous data. In joint training, all data are accessible and shuffled during the whole training process. The figure shows significant performance gap between the fine-tuning and the joint training strategies on downstream zero-shot classification task.

including zero-shot image classification and image-text retrieval [19]. However, training a CLIP model typically requires a large amount of image-text pairs (400 million), which is particularly burdensome for the traditional off-line training strategy as all the data need to be available during the entire training process. Moreover, for practical applications, it is critical for a VLP model to continuously integrate novel knowledge in a dynamic environment, e.g., from streaming data crawled from the Internet. On the other hand, as shown in Figure 1, a naive fine-tuning strategy for VLP using only the incoming new data suffers from a large performance degradation compared to the off-line training strategy. Consequently, it is essential to address this continual learning problem for large-scale VLPs, a topic that has received little attention in the past.

Traditional continual learning methods mostly focus on the issue of stability-plasticity dilemma [18,45], where a model is prone to forgetting the previously learned knowledge when adapting to new data. As such, much effort has been devoted to preserving discriminative features for the known classes [13]. For the VLP models, however, the pre-trained multi-modal representations need to be transferred to unseen downstream tasks. In this case, what knowledge should be retained and how it is preserved is less obvious. In addition, the forgetting problem in continual multi-modal learning involves the representation of visual and language inputs, as well as the multi-modal correspondence between the two modalities, which further complicates the problem.

Most state-of-the-art approaches [30] in traditional class incremental setting rely on the memory replay of representative old training samples [13]. Nevertheless, unlike the supervised class incremental learning [7] which typically trains a model to classify on a closed set of categories, VLP models learn an open set of visual concepts from natural language descriptions. Consequently, the majority of existing memory-based continual learning strategies would be rendered ineffective. In particular, those methods [52,47] aiming to alleviate class imbalance are not applicable to the continual learning of VLP models as there are no class-level supervision for VLP. In addition, the feature-based distillation methods [15] are usually designed for uni-modal CNNs and rarely take into account the characteristics of transformer-based multi-modal pre-training, and the model inversion [51] aims to generate images as positive examples for certain classes using a frozen copy of trained models, which remains challenging for high-resolution images and may introduce biases to the generated training data [46].

In this work, we propose a novel replay-based continual learning framework, named as IncCLIP, to address the above challenges for the VLP tasks. To this end, based on CLIP [34], we introduce a conditional data generation process that extracts the 'dark' knowledge from the previous-step model in a form of pseudo texts for replay, and adopt multi-modal knowledge distillation loss to further overcome catastrophic forgetting. Specifically, our model architecture is a two-stream encoder composed of separate visual and text encoders. At each incremental stage, in order to learn generic and transferable visual-linguistic representation, we adopt a contrastive loss that requires the model to predict the pairing between images and texts. Given the importance of negative instances in contrastive learning, we introduce a pseudo text generation technique via model inversion [51] to augment the data memory with informative data and replay the generated texts as negative examples. Moreover, to alleviate catastrophic forgetting, we design a knowledge distillation loss to minimize the output discrepancy between current and the previous-step model, which preserves the knowledge on cross-modal correspondence. It is also worth noting that after the incremental training of each step, we adopt reservoir sampling to update the memory for the rehearsal in the next step.

To validate our method, we perform continual model pre-training on an instance incremental and a class incremental split for the Conceptual Caption dataset [40], which simulates two different real scenarios. We then evaluate our model on two downstream tasks: zero-shot image classification and zero-shot image-text retrieval. The experimental results demonstrate the superiority of our approach, which is then further analyzed via the detailed ablation study. Notably, we outperform previous approaches by 4.6% in accuracy on the downstream image classification task with four-step class incremental split. In summary, the main contributions of our work are three-fold:

- To our best knowledge, this is the first work to tackle the problem of continual vision-language pre-training with streaming image-text pairs.
- To achieve better stability-plasticity trade-off, we propose the IncCLIP framework to augment the contrastive learning with the negative pseudo texts and adopt a multi-modal knowledge distillation loss between images and texts to preserve the learned cross-modal correspondence.
- Our proposed method consistently outperforms previous CL baselines such as UCIR in standard continual vision-language pre-training benchmarks.

2 Related Works

Vision-Language Pre-training Vision-language pre-training learns transferable joint image-text embeddings from large-scale image-text pairs, which has been shown to be effective for a variety of vision-language (VL) tasks [27,31,11]. The majority of existing works fall into two categories based on model architectures: single-stream and dual-stream models. Single-stream models [10,24,31] introduce powerful encoders such as Transformer [44] to model the cross-modal interaction between image and text. As such, they perform well on VL tasks like VQA [2], which requires complex reasoning between image and text. However, they typically use an external object detector to generate visual region descriptions as the input to the multi-modal encoder [11,10], which can be computationally expensive. Moreover, it is difficult to apply them to certain VL tasks such as image-text retrieval, which requires feeding all potential image-text pairs into the multi-modal encoder and hence is inefficient.

On the other hand, dual-stream models [23,34] adopt a dual-encoder architecture to encode images and texts, respectively. CLIP [34] and ALIGN [23] perform pre-training on large-scale noisy data collected from the Internet. Especially, CLIP provides a flexible zero-shot classifier rather than parametric task-specific classifiers, and demonstrates impressive zero-shot generalization ability on many downstream tasks, such as zero-shot image classification and zero-shot image-text retrieval. It is a significant step towards flexible and practical zero-shot classifiers for computer vision tasks. Nevertheless, current dual-stream VLP models are trained in a joint-training manner using data prepared in advance, without the ability to continuously adapt to new data from a dynamic environment. In this work, we concentrate on the continual vision-language representation learning based on dual-stream models like CLIP.

Continual Learning Continual learning [54,13,43,48] aims to integrate novel knowledge in a sequential fashion where old data are usually unavailable. Existing literature focuses mostly on supervised continual learning [50,53,41], which mainly falls into the following four groups. The first is the regularization methods [16,25], which penalize changes on significant weights of previously learned models. The second group is the distillation methods [28], which aim at retaining the output of the network on available data. The third is the structure methods [49,39], which keep old parameters fixed while growing and allocating weights for learning new data. The last is the pseudo rehearsal methods [51,42], which usually train a generative model to generate visual images of previously learned categories and train the classifier with the combination of real data and pseudo data to reduce forgetting. Our method combines the distillation method and the pseudo rehearsal method. However, previous pseudo rehearsal methods adopt either generative adversarial networks or variational auto-encoders, which are not easy to address their data degeneration issue, especially when dealing with complex scenarios such as high-resolution images or images of the similar classes. Our method circumvents this problem by "inverting" the model of last step to synthesize hard negative texts in the token embedding space, since token embedding has much lower dimensions and generating negative data points is easier. This is inspired by DeepInversion [51], but they do not consider generating negative data points and it is designed for the convolution network with Batch Normalization.

Recently, there also have been some efforts [36,22,8] to explore continual representation learning with unlabeled streaming data. CURL [36] proposes a continual unsupervised representation learning method, which learns a task-specific representation based on a set of share parameters and also trains a generative model to avoid forgetting. Co^{2}L[8] introduces a self-supervised knowledge distillation method for self-supervised continual learning. However, previous methods are designed for continual learning problems with a single modality and do not consider the properties of multi-modal representation learning. To the best of our knowledge, our work is the first to explore continual learning in the self-supervised multi-modality representation learning.

3 Methods

In this section, we describe our approach, as sketched in Figure 2, to address the continual vision-language representation learning problem, with the goal of improving stability-plasticity trade-off. Concretely, we combine contrastive loss and multi-modal knowledge distillation and supplement the training with pseudo texts to enhance the generalization ability for the representation. Below, we first present the overview of problem setup and model architecture in Sec. 3.1, followed by the introduction of text generation in Sec. 3.2. Then we detail the training loss in Sec. 3.3.

3.1 Problem Setup and Model Architecture

Problem Setup We first introduce the problem setup of **continual vision-language pre-training**. During sequential pre-training, the model observes a sequence of data chunks $\mathcal{D}_t$ for each step t. Particularly, the dataset $\mathcal{D}_t = \{(\boldsymbol{x}_i^I, \boldsymbol{x}_i^T)\}_{i=1}^{|\mathcal{D}_t|}$ is composed of image-text pairs at step t, where $\boldsymbol{x}_i^I$ means the input image, and $\boldsymbol{x}_i^T$ represents the corresponding text describing the image. We assess the transferability of learned representation to downstream tasks after the training at each chunk. In this work, all methods including our method and the comparison methods are based on the rehearsal strategy, which stores a subset of observed data in memory $\mathcal{M}_t = \{(\boldsymbol{x}_i^I, \boldsymbol{x}_i^T)\}$ for future training.

Model Architecture Like CLIP [34], we adopt a dual-stream encoder structure where the model $\mathcal{H}_t$ with parameters θ_t has independent visual encoder $f(\cdot)$ and text encoder $g(\cdot)$. Concretely, given an image $\boldsymbol{x}_i^I$ and a text $\boldsymbol{x}_j^T$, we first compute the normalized image embedding $\tilde{\boldsymbol{u}}_{\boldsymbol{i}}$ and normalized linguistic embedding $\tilde{\boldsymbol{v}}_{\boldsymbol{j}}$, and then compute the similarity score s_ij.

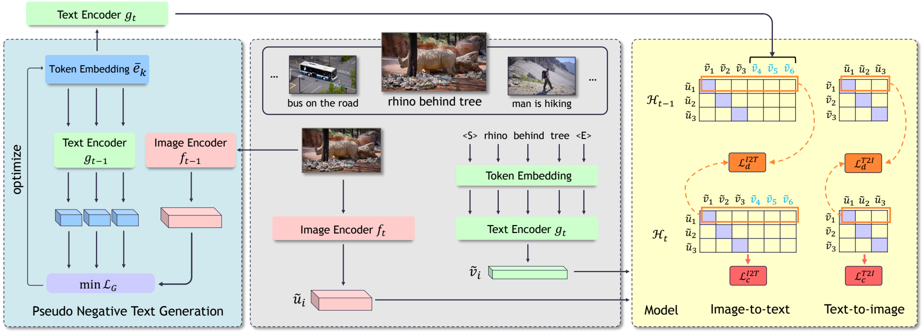

Fig. 2. The illustration of our method. The left panel illustrates the pseudo negative text generation process. We optimize pseudo texts $\bar{\boldsymbol{e}}_k$ in the token embedding space by minimizing the loss $\mathcal{L}_G$. As middle panel shows, for a sampled mini-batch, we first extract the images features $\tilde{\boldsymbol{u}}_i$ and language features $\tilde{\boldsymbol{v}}_i$ through the corresponding encoder f_t and g_t. The right panel shows the calculation of training loss. We compute the similarity matrix S between image features $\tilde{\boldsymbol{u}}_i$ and language features $\tilde{\boldsymbol{v}}_i$. Then we apply the contrastive loss $\mathcal{L}_c^{\mathrm{I2T}}$, $\mathcal{L}_c^{\mathrm{T2I}}$ and distillation loss $\mathcal{L}_d^{\mathrm{I2T}}$,$\mathcal{L}_d^{\mathrm{T2I}}$ from both image to text and text to image. Note that $\tilde{\boldsymbol{v}}_{\boldsymbol{i}}$ is the deep feature of the pseudo token embedding $\bar{\boldsymbol{e}}_{\boldsymbol{i}}$, which is used as negative examples in the training loss.

Concretely, to encode the image, we adopt ResNet as visual backbone, which extracts the features $\tilde{\boldsymbol{u}}_i = f(\boldsymbol{x}_i^I)$ from image $\boldsymbol{x}_i^I$, and normalize the feature $\tilde{\boldsymbol{u}}_i = \tilde{\boldsymbol{u}}_i / \|\tilde{\boldsymbol{u}}_i\|_2$ onto unit-sphere. To encode the text, we firstly tokenize the sequence into a sequence of word tokens $\boldsymbol{e}_i^T$ where we use the lower-cased byte pair encoding (BPE) [38] with a vocabulary size of 49,408 to tokenize the text. We adopt Transformer [44] as text encoder and encode these tokens into normalized linguistic embedding $\tilde{\boldsymbol{v}}_j = g(\boldsymbol{x}_j^T)/\|g(\boldsymbol{x}_j^T)\|_2$. Finally, we compute the similarity score $s_{ij} = \tilde{\boldsymbol{u}}_i^\top \tilde{\boldsymbol{v}}_j$ between i-th image $\boldsymbol{x}_i^I$ and j-th text $\boldsymbol{x}_j^T$.

3.2 Pseudo Negative Text Generation

In this subsection, we introduce the pseudo text generation via model inversion to distill the knowledge of last step model $\mathcal{H}_{t-1}$ for solving the catastrophic forgetting. Due to the distribution shift between synthetic and real examples, the model is frequently biased if regarding them as positive examples [42]. We bypass this issue via generating negative texts which do not guide the model incorrectly. Having access to informative negative samples is known to be critical for the success of contrastive learning [20]. Furthermore, it is observed that negative examples, especially hard negative examples, benefit the learning of representation [37]. Motivated by this, we propose to generate the negative text via model inversion as follows.

For each training batch, we perform the pseudo data generation to augment the mini-batch. Given the discrete nature of tokens, which makes the optimization difficult, we optimize the pseudo text $\bar{\boldsymbol{e}}_k$ in the token embedding space to find the hard negative texts with respect to the images $\boldsymbol{x}_i^I$. $\boldsymbol{e}_k$ should keep a moderate distance from $\boldsymbol{x}_i^I$ as a close distance indicates $\boldsymbol{e}_k$ is a positive sample and a remote distance means $\boldsymbol{e}_k$ is easy to be distinguished.

Specifically, to generate the pseudo texts, we firstly sample a mini-batch $\mathcal{B} = \{(\boldsymbol{x}_i^I, \boldsymbol{x}_i^T)\}_{i=1}^B$ from memory $\mathcal{M}_t$. We initialize the token embedding $\bar{\boldsymbol{e}}_k = \beta \boldsymbol{e}_i^T + (1-\beta)\boldsymbol{e}_j^T$ where $\bar{\boldsymbol{e}}_k$ are k-th generated text, β is uniformly sampled from the interval $[0, 1]$, i, j are indices randomly sampled from 1 to batch size B, and $\boldsymbol{e}_i^T$, $\boldsymbol{e}_j^T$ are the corresponding token embeddings. It is worth noting that $\bar{\boldsymbol{e}}_k$ takes the value of a real token embedding as initialization when $\beta = 0$ or $\beta = 1$. Given the current pseudo token embedding $\bar{\boldsymbol{e}}_k$, we compute the cosine similarity scores s_{ik} with the images features $\tilde{\boldsymbol{u}}_i$. Then we generate the data in the token embedding space which is continuous and easy to optimize compared to discrete tokenization space. The naive text generation via model inversion is to directly minimize the cosine similarity, which are calculated by the inner product between the generated token sequences $\bar{\boldsymbol{e}}_k$ and the sampled image features $\tilde{\boldsymbol{u}}_i$. However, to improve data efficiency, we require the generated token embedding $\bar{\boldsymbol{e}}_k$ to be hard negative examples which means they are difficult to be distinguished from positive examples. To achieve this, we adopt margin loss as follows

$$\mathcal{L}_G = \frac{1}{B}\sum_{i=1}^{B} \max(0, s_{\min} - s_{ik}) + \max(0, s_{ik} - s_{\max}), \tag{1}$$

where $s_{\min}$ and $s_{\max}$ are the hyper-parameters representing the minimum and maximum score, respectively. The maximum score guarantees the generated example to be negative examples. The minimum score requires the generated examples to be hard negative examples. We adopt SGD to minimize the loss $\mathcal{L}_G$ with respect to the variable $\bar{\boldsymbol{e}}_k$ for a constant number of iterations.

3.3 Training Loss

We now describe the details of our continual pre-training objective. For pretraining, we adopt a contrastive learning task to learn a generic and transferable visual-linguistic representation, which has proven to be efficient and effective in previous works [27]. To preserve the knowledge in the model, we maintain the relation of instances by introducing the knowledge distillation. With the above generated texts $\bar{\boldsymbol{e}}_k$, we use it in both contrastive learning and multi-modal knowledge distillation, which helps the contrastive learning on novel data and improves the regularization of distillation loss.

During continual learning, we learn the model on the union of incoming data, memory and pseudo texts. In detail, we sample a mini-batch of image-text pairs $\{(\boldsymbol{x}_i^I, \boldsymbol{x}_i^T)\}_{i=1}^B$ from incoming data $\mathcal{D}_t$ and memory $\mathcal{M}_t$ where B denotes the batch size, and sample a mini-batch of token embedding $\{\boldsymbol{e}_k^T\}_{k=1}^{\hat{B}}$ of negative texts with batch size $\hat{B}$. We denote the text batch size in total as $B_T = B + \hat{B}$. The linguistic embedding for the sampled texts $\boldsymbol{x}_i^T$ and the token embedding $\boldsymbol{e}_k^T$ of negative texts are denoted as $\{\tilde{\boldsymbol{v}}_j\}_{j=1}^{B_T}$, and the visual embedding are

$\{\tilde{\boldsymbol{u}}_i\}_{i=1}^{B}$. We can compute the classification probability $P_{I2T}(y_i|\tilde{\boldsymbol{u}}_i, \{\tilde{\boldsymbol{v}}_j\}_{j=1}^{B_T})$, $y_i \in \{1, 2, \ldots, B_T\}$, from image to texts to determine which text corresponds to the given image as follows

$$P_{I2T}(y_i = k|\tilde{\boldsymbol{u}}_i, \{\tilde{\boldsymbol{v}}_j\}_{j=1}^{B_T}; \tau) = \frac{\exp(s_{ik}/\tau)}{\sum_{j=1}^{B_T} \exp(s_{ij}/\tau)}, \tag{2}$$

where τ is the temperature parameter to control the smoothness of the Softmax function, and $P_{I2T}(y_i = k|\tilde{\boldsymbol{u}}_i, \{\tilde{\boldsymbol{v}}_j\}_{j=1}^{B_T}; \tau)$ means the chance of i-th image being paired with k-th text. Similarly, we can compute the classification probability $P_{T2I}(y_j|\tilde{\boldsymbol{v}}_j, \{\tilde{\boldsymbol{u}}_i\}_{i=1}^{B})$ from text to images, i.e. the chance of j-th text being paired with k-th image, as follows

$$P_{T2I}(y_j = k|\tilde{\boldsymbol{v}}_j, \{\tilde{\boldsymbol{u}}_i\}_{i=1}^{B}; \tau) = \frac{\exp(s_{kj}/\tau)}{\sum_{i=1}^{B} \exp(s_{ij}/\tau)}. \tag{3}$$

Contrastive Loss We adopt bi-directional contrastive loss to learn generalized image representations from natural language supervision. We jointly train an image encoder and a text encoder to predict the correct pairings of a batch of image-text pairs. Concretely, for an image $\boldsymbol{x}_i^I$, we regard its corresponding language description $\boldsymbol{x}_i^T$ as a positive example whereas the other $B_T - 1$ texts are considered negative examples. Therefore, the image-to-text loss is as follows

$$\mathcal{L}_c^{I2T} = -\frac{1}{B}\sum_{i=1}^{B} \log P_{I2T}(y_i|\tilde{\boldsymbol{u}}_i, \{\tilde{\boldsymbol{v}}_j\}_{j=1}^{B_T}; \tau), \tag{4}$$

where the ground truth $y_i \in \{1, 2, \ldots, B\}$. The text-to-image loss $\mathcal{L}_c^{T2I}$ is defined on the texts $\{\boldsymbol{x}_i^T\}_{i=1}^{B}$ in a similar way as follows

$$\mathcal{L}_c^{T2I} = -\frac{1}{B}\sum_{i=1}^{B} \log P_{T2I}(y_j|\tilde{\boldsymbol{v}}_j, \{\tilde{\boldsymbol{u}}_i\}_{i=1}^{B}; \tau), \tag{5}$$

where we only compute the loss on real texts because the pseudo texts lack a paired image. In total, the overall contrastive loss $\mathcal{L}_c$

$$\mathcal{L}_c = \alpha\mathcal{L}_c^{I2T} + (1-\alpha)\mathcal{L}_c^{T2I}, \tag{6}$$

where hyper-parameter α is the loss weighting coefficient.

Cross-modal Knowledge Distillation To prevent catastrophic forgetting, knowledge distillation is introduced to keep the instance-wise prediction between current model $\mathcal{H}_t$ and the model $\mathcal{H}_{t-1}$ learned at last task. Concretely, we retain an image's relationships with texts by employing the knowledge distillation loss

$\mathcal{L}_d^{I2T}$ with KL-divergence as follows

$$\begin{aligned}\mathcal{L}_d^{I2T} = \frac{1}{B}\sum_{i=1}^{B} \mathrm{KL}\Big(& P_{I2T}(y_i|\tilde{\boldsymbol{u}}_i, \{\tilde{\boldsymbol{v}}_j\}_{j=1}^{B_T}; \theta_t, \tau)|| \\ & P_{I2T}(y_i|\tilde{\boldsymbol{u}}'_i, \{\tilde{\boldsymbol{v}}'_j\}_{j=1}^{B_T}; \theta_{t-1}, \tau^d_{old})\Big),\end{aligned} \tag{7}$$

where $\tilde{\boldsymbol{u}}'_i$,$\tilde{\boldsymbol{v}}'_j$ are the features extracted the model $\mathcal{H}_{t-1}$, τ^d_{old} represents the temperatures of last model for distillation. Similarly, for a text $\boldsymbol{x}_j^T$, $1 \leq j \leq B$, we also apply distillation loss on the prediction probabilities from text to image as follows

$$\begin{aligned}\mathcal{L}_d^{T2I} = \frac{1}{B}\sum_{j=1}^{B} \mathrm{KL}\Big(& P_{T2I}(y_j|\tilde{\boldsymbol{u}}_j, \{\tilde{\boldsymbol{v}}_i\}_{i=1}^{B}; \theta_t, \tau)|| \\ & P_{T2I}(y_j|\tilde{\boldsymbol{u}}'_j, \{\tilde{\boldsymbol{v}}'_i\}_{i=1}^{B}; \theta_{t-1}, \tau^d_{old})\Big).\end{aligned} \tag{8}$$

Therefore, the knowledge distillation loss

$$\mathcal{L}_d = \eta\mathcal{L}_d^{I2T} + (1-\eta)\mathcal{L}_d^{T2I}, \tag{9}$$

where the hyper-parameter η is the loss weight.

Overall Loss Finally, we combine the contrastive loss $\mathcal{L}_c$ and distillation loss $\mathcal{L}_d$, and obtain the final loss as follows

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{c}} + \lambda\mathcal{L}_{\text{d}}, \tag{10}$$

where λ is the loss coefficient to control the tradeoff between losses.

Notably, we find that the weight norm often increases over different steps, which hampers the generalization ability of the pretrained model and causes negative forward transfer. The same phenomenon is also observed in recent works [3]. Empirically, we adopt the trick of weight norm clipping. Concretely, at the end of each training iteration, if the weight norm at layer-l is higher than δ_l, we clip the weight norm to δ_l when keeping the direction of the weight W_l unchanged. In practice, $\delta_l = \gamma\|W\|_{\text{init}}$ where the initial weight norm are denoted as $\|W\|_{\text{init}}$. After the training of step t, we follow the practices [1,9] to update the memory by adopting reservoir sampling to select samples from the avaiable data to save.

4 Experiments

In this section, we conduct exhaustive experiments to validate the effectiveness of our method. Concretely, we first describe the experimental setup and implementation details in Sec. 4.1, followed by the evaluation results on class incremental split in Sec. 4.2. Then we introduce the experimental results on the instance incremental split in Sec. 4.3. Finally, we perform ablation study and analysis to validate the effectiveness of components and provide more insights in Sec. 4.4.

Table 1. Results on class incremental CC2M dataset at final step: The top-1 accuracy over various downstream datasets on zero-shot image classification task.

#Tasks	Methods	ImageNet	CIFAR-10	CIFAR-100	Caltech101	SUN397	Food101	Flowers102	DTD	Average
	Joint	29.97	51.94	26.04	65.2	31.79	23.71	19.44	12.82	32.61
4	ER [9]	14.45	23.56	7.84	35.93	16.83	12.59	10.17	8.94	16.29
	UCIR [21]	13.24	25.56	8.47	35.14	17.13	13.05	9.97	**9.84**	16.55
	Co^{2}L [8]	14.73	26.46	10.51	34.58	17.54	12.16	11.41	7.3	16.84
	GeoDL [41]	14.24	27.48	9.49	35.93	17.01	12.94	9.87	8.99	17.00
	IncCLIP	**18.85**	**28.31**	**13.23**	**50.32**	**23.38**	**16.19**	**13.08**	9.20	**21.57**
8	ER [9]	9.59	14.23	3.85	26.80	11.95	7.24	8.98	5.48	11.02
	UCIR [21]	9.89	12.69	4.42	23.56	12.95	9.53	9.33	7.02	11.17
	Co^{2}L [8]	10.99	18.34	5.51	29.1	13.52	9.01	8.56	5.9	12.62
	GeoDL [41]	10.86	14.88	5.11	30.56	14.2	10.17	8.09	7.18	12.64
	IncCLIP	**13.93**	**22.68**	**10.45**	**43.39**	**19.27**	**11.91**	**9.57**	**7.38**	**17.24**

4.1 Experiment Setup and Implementation Details

Benchmark Protocol Conceptual 12M (CC12M) [40] is a dataset collected from the Internet including 12 million image-text pairs for vision-language pretraining. In this section, we show results on both the class incremental and instance incremental split, corresponding to large and insignificant distribution shift in real world, respectively. Considering that the image labels are not provided, we adopt an approximate strategy to build our class incremental split. Specifically, we first generate a pseudo class label for each image and then partition ImageNet1k dataset into four chunks with 250 classes per chunk. Here the pseudo labels are the most confident predictions from the BEiT [4] model pretrained on ImageNet1k. For the instance incremental split, it models a real-world scenario in which data chunks are continuously collected in the same environment [22]. In particular, we build the split via randomly selecting 2M image-text pairs from CC12M and then randomly split them into 4 identically distributed chunks with 0.5M image-text pairs each chunk. Note that the 2M image-text pairs used for instance incremental setting here is different with image-text pairs of the class incremental split for verifying the robustness of algorithm. For completeness, we provide experiments for instance incremental split on same 2M image-text pairs with class incremental split in appendix. The splits will be released in the future. We allow the algorithms to store fixed-size instances in memory.

Comparison Methods To demonstrate the efficacy of our method, we adopt ER [9], UCIR [21], GeoDL [41] as comparison methods. Notably, we replace the

Table 2. Image-text Retrieval Performance at final step: Zero-shot Image-Text Retrieval on MSCOCO and Flickr30k datasets with various methods. R@K means top-K recall.

#Tasks	Methods	Flickr30K						MSCOCO					
		image-to-text			text-to-image			image-to-text			text-to-image		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
	Joint	35.7	62.4	71.8	25.16	50.52	62.36	18.8	41.82	52.94	13.14	30.74	41.78
4	ER [9]	17.5	39.8	51.3	11.18	28.68	38.18	9.6	24.72	35.34	6.41	17.76	25.61
	GeoDL [41]	18.1	40.4	51.1	11.96	29.94	39.4	9.64	25.58	35.78	6.26	18.02	26.01
	UCIR [21]	18.3	41.9	52.9	12.10	30.32	40.28	9.72	25.88	36.16	6.64	18.58	26.83
	Co^{2}L [8]	19.7	42.5	53.1	12.24	29.34	39.54	10.1	25.16	35.24	6.78	18.30	26.59
	IncCLIP	**24.1**	**49.5**	**61.9**	**17.14**	**37.96**	**48.96**	**12.38**	**29.96**	**40.6**	**8.49**	**22.55**	**31.90**
8	ER [9]	9.7	27.3	38.5	6.54	19.16	27.7	6.47	16.84	24.48	4.22	13.01	19.05
	GeoDL [41]	12.5	33.3	42.1	8.40	21.9	30.44	6.42	18.08	27.32	4.64	14.02	20.58
	UCIR [21]	10.1	28.7	40.1	7.16	20.54	29.14	7.00	17.58	25.22	4.38	13.13	19.83
	Co^{2}L [8]	12.9	32.3	41.9	8.43	21.76	30.41	6.22	18.68	26.58	4.49	13.29	20.24
	IncCLIP	**16.0**	**37.7**	**49.2**	**10.92**	**28.26**	**38.60**	**9.52**	**24.18**	**33.60**	**6.29**	**17.41**	**26.08**

Table 3. Ablation Study: W.N.C means the weight norm cliping, H.N.T.G means hard negative text generation and Dist. means the knowledge distillation loss on image classification and image-to-text retrieval task. Below results on image-to-text retrieval tasks are top-1 recall.

W.N.C	Dist.	H.N.T.G	ImageNet	Flickr30K		MSCOCO	
				I2T	T2I	I2T	T2I
✓	✗	✗	16.01	21.3	14.06	10.89	7.52
✓	✓	✗	17.67	22.8	15.10	11.55	8.02
✓	✓	✓	**18.85**	**24.1**	**17.14**	**12.38**	**8.49**

cross entropy loss with contrastive loss to train a VLP model for ER, UCIR and GeoDL. Besides, we also provide the joint-training(Joint) as an upper bound which makes use of all observed data to update the model.

Implementation Details Following CLIP [34], all methods including comparison methods and our method IncCLIP adopt Transformer [44] as the text backbone with the architecture modifications described in [35]. Particularly, we use 12-layer 512-wide transformer with 8 attention heads and modified ResNet-50 [34]. The results on zero-shot image classification are reported using the prompt ensemble technique [34]. For each mini-batch, the size of sampled negative text $B_{\text{aug}} = 256$. The loss coefficient $\lambda = 5$. $\alpha = 0.5$, $\eta = 0.5$, $\tau^d_{old} = 0.01$. The following experiments are conducted with 10% dataset as memory except otherwise stated. To train our model, we adopt 8 Nvidia V100 GPUs with batch size 512 per GPU. For each step, we train the model for 15 epochs on CC2M.We adopt LAMB optimizer with learning rate 0.003 and weight decay 0.003. We begin by performing a linear warmup at each incremental step and then decay it using a cosine learning rate schedule. More details can be found in appendix.

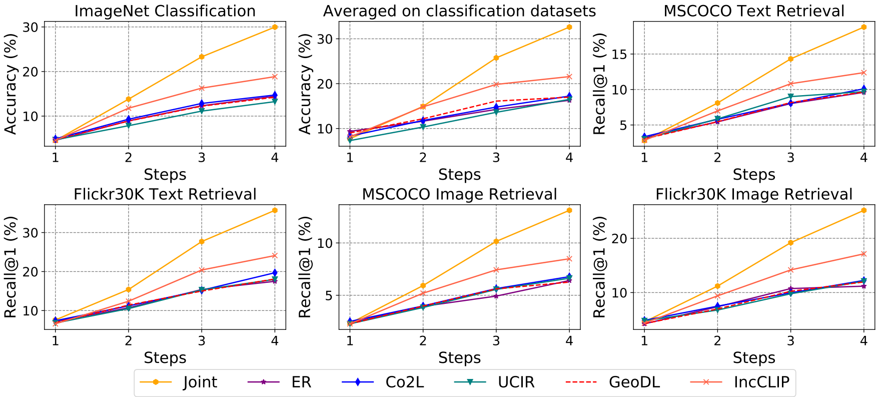

Fig. 3. The downstream task performance over time. For image classification task, the accuracy is determined by averaging accuracies of all downstream datasets.

Table 4. The sensitive study of memory size for image classification task. 'Average' means the average accuracy over all downstream image classification datasets.

	ImageNet					Average				
	1%	5%	10%	20%	50%	1%	5%	10%	20%	50%
IncCLIP	12.82	16.68	18.95	23.61	28.29	16.90	20.16	21.82	24.92	30.61

4.2 Class Incremental Split

Zero-shot Image Classification We conduct experiments to evaluate the algorithm's zero-shot generalization ability on image classification task. Concretely, we evaluate models on eight representative datasets including CIFAR-10 [26], CIFAR-100 [26], Caltech101 [17], Oxford 102 Flower(Flowers102) [32], Food101 [6], SUN397 [5], Describable Textures Dataset (DTD) [12], and ImageNet [14]. Like CLIP [34], we adopt prompt template, embed the class name to acquire the prediction score for each class, then use the class with the highest score as the prediction label during inference.

Table 1 summarizes the final step performance of CC2M dataset with 4 and 8 steps. We can see that our method regularly surpasses other methods with a significant margin at different downstream datasets. Specifically, our method improves the accuracy from 19.7 to 24.1(+**4.4**%) under the 4 step split on ImageNet. It is observed that when the number of steps increases, the average gain from our method increases as well. In spite of the fact that traditional methods like UCIR,Co^{2}L are better than ER , it can be seen that the improvement is limited compared to ER, indicating that their direct application can not fix the problem well. Moreover, although our method achieves obvious gain, the large gap between our method and Joint(Upper bound) indicates that continual vision-language pre-training is challenging to be solved. Furthermore, despite the clear increase achieved by our method, the wide gap between our method and

Table 5. The sensitive study on memory size for image-to-text retrieval on Flickr30K.

	image-to-text					text-to-image				
	1%	5%	10%	20%	50%	1%	5%	10%	20%	50%
IncCLIP	18.2	21.6	24.1	27.2	31.6	13.16	15.58	17.14	18.38	23.62

the upper bound 'Joint' suggests that continuous vision-language pre-training is far from to be solved. As shown in the Figure 3, it is observed that our method consistently outperforms the comparison methods at different steps.

Zero-shot Image-Text Retrieval The image-text retrieval task consists of two sub-tasks: image-to-text retrieval and text-to-image retrieval. In particular, we employ MSCOCO [29] and Flickr30K [33] datasets to assess the representation transferability of pretrained representation on image-text retrieval task. Table 2 demonstrates the image-text retrieval results on 4-step split of CC2M. We can see that our method consistently outperforms the other methods in both image-to-text and text-to-image retrieval tasks. Particularly, our method improves from 10.1% to 12.38% for top-1 text recall on MSCOCO dataset at final step. As the number of steps increases from 4 to 8, our method's performance on all metrics falls, indicating that the continuous vision-language pre-training task becomes more difficult for longer sequences.

4.3 Instance Incremental Split

As Table 6 shows, we evaluate the methods on instance incremental split, and our method consistently outperforms than other methods on both classification and retrieval tasks tasks, showing the superiority and the robustness of our method. Our method achieves 2.12% improvement on ImageNet classification task.

4.4 Ablation Study and Analysis

Ablation Study We conduct exhaustive ablation study to evaluate the influence of each component used in our method. As shown in Table 3, we can see that the generalization performance is improved when weight norm clipping is applied. Moreover, it is observed that the introduction of knowledge distillation loss gains 1.66% improvement on classification accuracy on ImageNet. Finally, with the addition of negative text replay, we can further obtain 1.3% top-1 text recall performance gain on Flickr30K dataset for text retrieval task.

Sensitive Study on Memory Size We conduct sensitive study on memory size and report the results in Table 4 and 5. '%x' indicates that we set the memory size to be x percents of the total size of the dataset. We can find that the performance on all tasks consistently improves as the memory size increases, which demonstrates the effectiveness of replay strategy once more.

Table 6. Results on instance incremental split of 2M image-text pairs at final step: 'Average' means the accuracy averaged on eight classification datasets. Moreover, we report the top-1 recall on both MSCOCO and Flickr30K dataset.

Methods	ImageNet	Average	Flickr30K		MSCOCO	
			I2T	T2I	I2T	T2I
Joint	20.20	24.58	27.20	19.02	14.58	9.90
ER [9]	10.74	14.38	16.3	10.74	8.50	5.49
UCIR [21]	10.57	14.91	16.7	11.34	9.14	6.11
GeoDL [41]	10.85	14.16	16.4	10.82	8.62	5.65
Co^{2}L [8]	11.12	15.33	18.3	10.86	8.58	5.69
IncCLIP	**13.24**	**17.97**	**26.5**	**17.18**	**13.46**	**8.60**

Table 7. The forgetting analysis on various methods.

Methods	ER	UCIR	Co^{2}L	GeoDL	IncCLIP
BWT	-31.93	-28.68	-30.83	-31.41	-18.48

Forgetting Analysis Additionally, we conduct experiments to analyze the algorithm's resistance to forgetting. To measure the forgetting in the continual vision-language pre-training, we define the backward transfer(BWT) as the accuracy changes on each training chunk. Detailedly, BWT $= \frac{1}{N-1}\sum_{i=2}^{N}\frac{1}{i}\sum_{j=1}^{i}A_j^i - A_j^j$ where A_j^i means the accuracy of step i model $\mathcal{H}_i$ on chunk $j \leq i$. The model has forgotten part of knowledge it acquired in chunk j when $A_j^i - A_j^j \leq 0$. As Table 7 shows, we can see that our method is less prone to catastrophic forgetting, which indicates the superiority of our method.

5 Conclusions

In this work, we have proposed a novel continual learning problem to learn generic and transferable vision-language representation. To overcome the catastrophic forgetting, we develop a replay-based framework with two main contributions. First, we perform model inversion to generate hard negative texts in token embedding space conditioned on the available images, and then use them to augment training. Second, we adopt contrastive learning as our pre-training objective and introduce the knowledge distillation on the similarity scores between images and texts. We conduct extensive experiments on Conceptual Caption dataset, and show that our learning strategy outperforms previous methods. While the performance of our method is promising, the text generation module requires extra training time, which can be improved in future work.

Acknowledgements This work was supported by Shanghai Science and Technology Program 21010502700, and Shanghai Frontiers Science Center Program. This work was done when Shipeng Yan was intern at Noah's Ark Lab. Hang Xu and Xuming He are equal corresponding authors.

References

1. Aljundi, R., Belilovsky, E., Tuytelaars, T., Charlin, L., Caccia, M., Lin, M., Page-Caccia, L.: Online continual learning with maximal interfered retrieval. In: NeurIPS
2. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: VQA: Visual question answering. In: ICCV (2015)
3. Ash, J., Adams, R.P.: On warm-starting neural network training. In: NeurIPS (2020)
4. Bao, H., Dong, L., Wei, F.: Beit: Bert pre-training of image transformers. arXiv:2106.08254 (2021)
5. Barriuso, A., Torralba, A.: Notes on image annotation. arXiv:1210.3448 (2012)
6. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101 – mining discriminative components with random forests. In: ECCV (2014)
7. Castro, F.M., Marín-Jiménez, M.J., Guil, N., Schmid, C., Alahari, K.: End-to-end incremental learning. In: ECCV (2018)
8. Cha, H., Lee, J., Shin, J.: Co2l: Contrastive continual learning. In: CVPR (2021)
9. Chaudhry, A., Rohrbach, M., Elhoseiny, M., Ajanthan, T., Dokania, P.K., Torr, P.H., Ranzato, M.: On tiny episodic memories in continual learning. arXiv:1902.10486 (2019)
10. Chen, Y.C., Li, L., Yu, L., El Kholy, A., Ahmed, F., Gan, Z., Cheng, Y., Liu, J.: Uniter: Universal image-text representation learning. In: ECCV (2020)
11. Cho, J., Lei, J., Tan, H., Bansal, M.: Unifying vision-and-language tasks via text generation. arXiv:2102.02779 (2021)
12. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. In: CVPR (2014)
13. Delange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., Tuytelaars, T.: A continual learning survey: Defying forgetting in classification tasks. TPAMI (2021)
14. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
15. Douillard, A., Cord, M., Ollion, C., Robert, T., Valle, E.: Podnet: Pooled outputs distillation for small-tasks incremental learning. In: ECCV (2020)
16. Ebrahimi, S., Elhoseiny, M., Darrell, T., Rohrbach, M.: Uncertainty-guided continual learning with bayesian neural networks. In: ICLR (2019)
17. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. TPAMI (2006)
18. Grossberg, S.: Adaptive resonance theory: How a brain learns to consciously attend, learn, and recognize a changing world. Neural Networks (2013)
19. Han, Z., Fu, Z., Chen, S., Yang, J.: Contrastive embedding for generalized zero-shot learning. In: CVPR (2021)
20. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR (2020)
21. Hou, S., Pan, X., Loy, C.C., Wang, Z., Lin, D.: Learning a unified classifier incrementally via rebalancing. In: CVPR (2019)
22. Hu, D., Yan, S., Lu, Q., Hong, L., Hu, H., Zhang, Y., Li, Z., Wang, X., Feng, J.: How well self-supervised pre-training performs with streaming data? In: ICLR (2022)
23. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q.V., Sung, Y., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. arXiv:2102.05918 (2021)

24. Kim, W., Son, B., Kim, I.: Vilt: Vision-and-language transformer without convolution or region supervision. In: ICML (2021)
25. 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. Proceedings of the National Academy of Sciences (PNAS) (2017)
26. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Technical Report (2009)
27. Li, J., Selvaraju, R.R., Gotmare, A.D., Joty, S., Xiong, C., Hoi, S.: Align before fuse: Vision and language representation learning with momentum distillation. In: NeurIPS (2021)
28. Li, Z., Hoiem, D.: Learning without forgetting. TPAMI (2017)
29. Lin, T.Y., Maire, M., Belongie, S.J., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
30. Liu, Y., Schiele, B., Sun, Q.: Meta-aggregating networks for class-incremental learning. arXiv:2010.05063 (2020)
31. Liu, Y., Wu, C., Tseng, S.y., Lal, V., He, X., Duan, N.: Kd-vlp: Improving end-to-end vision-and-language pretraining with object knowledge distillation. arXiv preprint arXiv:2109.10504 (2021)
32. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. Computer Graphics and Image Processing (2008)
33. Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. IJCV (2017)
34. 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: ICML (2021)
35. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI Blog (2019)
36. Rao, D., Visin, F., Rusu, A., Pascanu, R., Teh, Y.W., Hadsell, R.: Continual unsupervised representation learning. In: NeurIPS (2019)
37. Robinson, J.D., Chuang, C.Y., Sra, S., Jegelka, S.: Contrastive learning with hard negative samples. In: ICLR (2021)
38. Sennrich, R., Haddow, B., Birch, A.: Neural machine translation of rare words with subword units. arXiv:1508.07909 (2015)
39. Serra, J., Suris, D., Miron, M., Karatzoglou, A.: Overcoming catastrophic forgetting with hard attention to the task. In: ICML (2018)
40. Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: ACL (2018)
41. Simon, C., Koniusz, P., Harandi, M.: On learning the geodesic path for incremental learning. In: CVPR (2021)
42. Smith, J., Hsu, Y.C., Balloch, J., Shen, Y., Jin, H., Kira, Z.: Always be dreaming: A new approach for data-free class-incremental learning. arXiv:2106.09701 (2021)
43. Thrun, S.: Lifelong learning algorithms. In: Learning to learn (1998)
44. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
45. Wang, L., Yang, K., Li, C., Hong, L., Li, Z., Zhu, J.: OrdisCo: Effective and efficient usage of incremental unlabeled data for semi-supervised continual learning. In: CVPR (2021)

46. Wang, S.Y., Wang, O., Zhang, R., Owens, A., Efros, A.A.: Cnn-generated images are surprisingly easy to spot... for now. In: CVPR (2020)
47. Wu, Y., Chen, Y., Wang, L., Ye, Y., Liu, Z., Guo, Y., Fu, Y.: Large scale incremental learning. In: CVPR (2019)
48. Xie, J., Yan, S., He, X.: General incremental learning with domain-aware categorical representations. In: CVPR (2022)
49. Yan, S., Xie, J., He, X.: DER: Dynamically expandable representation for class incremental learning. In: CVPR (2021)
50. Yan, S., Zhou, J., Xie, J., Zhang, S., He, X.: An em framework for online incremental learning of semantic segmentation. In: ACM MM (2021)
51. Yin, H., Molchanov, P., Alvarez, J.M., Li, Z., Mallya, A., Hoiem, D., Jha, N.K., Kautz, J.: Dreaming to distill: Data-free knowledge transfer via deepinversion. In: CVPR (2020)
52. Zhao, B., Xiao, X., Gan, G., Zhang, B., Xia, S.T.: Maintaining discrimination and fairness in class incremental learning. In: CVPR (2020)
53. Zhao, H., Qin, X., Su, S., Fu, Y., Lin, Z., Li, X.: When video classification meets incremental classes. In: ACM MM (2021)
54. Zhao, H., Wang, H., Fu, Y., Wu, F., Li, X.: Memory efficient class-incremental learning for image classification. TNNLS (2021)