Preventing Zero-Shot Transfer Degradation in Continual Learning of Vision-Language Models

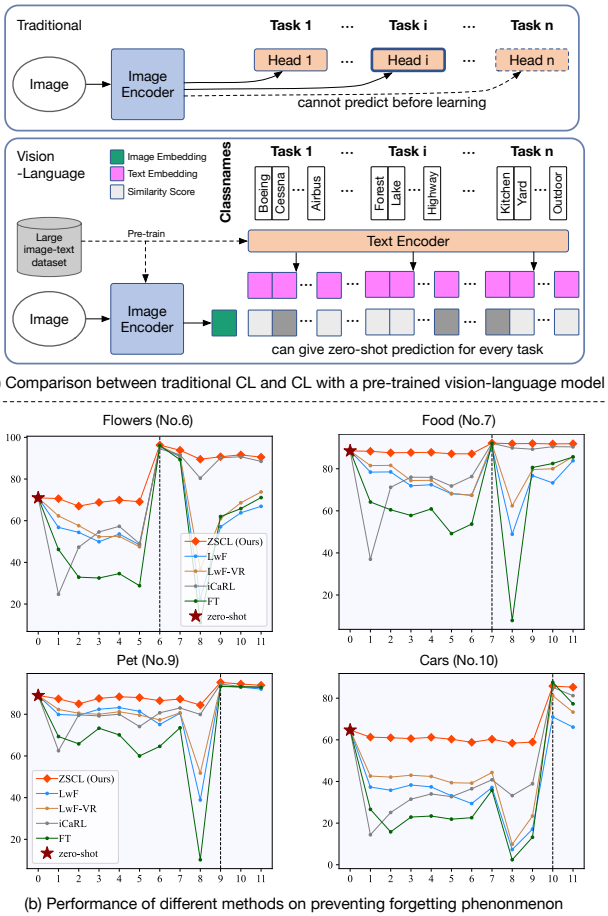
Zangwei Zheng[1] Mingyuan Ma[2] Kai Wang[1] Ziheng Qin[1] Xiangyu Yue[3] Yang You[1]
[1]National University of Singapore [2]UC Berkeley [3]The Chinese University of Hong Kong
[1]{zangwei, kai.wang, zihengq, youy}@comp.nus.edu.sg [2]mamingyuan2001@berkeley.edu [3]xyyue@ie.cuhk.edu.hk

Abstract

Continual learning (CL) can help pre-trained vision-language models efficiently adapt to new or under-trained data distributions without re-training. Nevertheless, during the continual training of the Contrastive Language-Image Pre-training (CLIP) model, we observe that the model's zero-shot transfer ability significantly degrades due to catastrophic forgetting. Existing CL methods can mitigate forgetting by replaying previous data. However, since the CLIP dataset is private, replay methods cannot access the pre-training dataset. In addition, replaying data of previously learned downstream tasks can enhance their performance but comes at the cost of sacrificing zero-shot performance. To address this challenge, we propose a novel method ZSCL to prevent zero-shot transfer degradation in the continual learning of vision-language models in both feature and parameter space. In the feature space, a reference dataset is introduced for distillation between the current and initial models. The reference dataset should have semantic diversity but no need to be labeled, seen in pre-training, or matched image-text pairs. In parameter space, we prevent a large parameter shift by averaging weights during the training. We propose a more challenging Multi-domain Task Incremental Learning (MTIL) benchmark to evaluate different methods, where tasks are from various domains instead of class-separated in a single dataset. Our method outperforms other methods in the traditional class-incremental learning setting and the MTIL by 9.7% average score. Our code locates at https://github.com/Thunderbeee/ZSCL.

1. Introduction

Most deep learning models can access all the data during training [27, 16, 12]. If we want to expand a model's knowledge, such as learning a newly found animal species [41], we can re-train the model by adding new classes to the training dataset. However, re-training a large model is costly. In contrast, continual learning (CL) [33, 47] incrementally learns task one after another. It can reduce



(a) Comparison between traditional CL and CL with a pre-trained vision-language model

(b) Performance of different methods on preventing forgetting phenonmenon

Figure 1. a) Conventional CL learns distinct task-specific heads, while CL with vision-language models can predict both learned tasks and out-of-distribution tasks. b) Accuracy (%) changes during CL of four datasets on 11 datasets. Our method is superior to others in preventing the forgetting of both zero-shot transfer ability and new knowledge.

this cost by only learning the new data, thereby presenting itself as an efficient alternative to conventional learning methods [43, 34, 5]. Nonetheless, a model tends to forget previous information catastrophically when learning new tasks [47, 33, 59]. The "catastrophic forgetting" phenomenon is a great challenge for CL.

Recently, vision-language models have shown powerful zero-shot transfer ability [44, 23, 32]. They can give zero-shot predictions without any training examples of a task. However, the performance on some tasks is poor due to insufficient relevant image-text pairs in the pre-training datasets. For example, it is difficult for CLIP [44] to distinguish among digital numbers, with an accuracy on MNIST [8] below 60% much lower than a naively trained CNN [29]. If we want to widen the knowledge in the vision-language model by re-training, the computational cost is too large (e.g., CLIP is pretrained on 400 million image-text pairs). Fine-tuning downstream tasks achieves high performance, but one model for a task takes much memory, and the model is not reusable. Prompt learning [64, 63] keeps the backbone parameters unchanged. However, it is only effective with limited training data due to a limited prompt length [64, 24]. In contrast, continual learning makes learning new knowledge a lifelong process for the vision-language model. The continually learned model can handle any image-text input and can be further used for downstream tasks [11, 53].

We find that existing CL methods hardly prevent the forgetting phenomenon for zero-shot transfer ability in continual learning of a pre-trained vision-language model. As shown in Fig. 1 (a), the CL with a pre-trained vision-language model differs from the traditional one. Besides forgetting previously learned task knowledge, the CLIP-based CL suffers from forgetting pre-training knowledge, namely a degradation of zero-shot transfer ability. For the replay-based CL methods [47, 50, 36, 21, 28, 42], the dataset during pre-training may be private and inaccessible during fine-tuning. For distillation-based CL methods [33, 10, 13, 11], they do not lay enough emphasis on the pre-trained model. On the one hand, a large model state change hinders tasks thereafter from using high-quality feature representations. On the other hand, it significantly degrades zero-shot performance on unseen datasets.

Our method ZSCL protects the Zero-Shot transfer ability during Continual Learning. We view the knowledge stored in the pre-training model from two perspectives: a well-learned feature space and a good value in the parameter space. In feature space, we re-design previous distillation loss [18, 47] with different loss styles, teacher models, and data sources. We find the original CLIP model, as opposed to the newly acquired model, is the best option for the teacher model. Instead of using data collected from previous tasks [47] or current task [18], we find a reference dataset with diverse semantics (e.g., images sampled from ImageNet) is a good option for distillation loss. The reference images need not be labeled or matched with the text. Preserving the relative similarity between reference images and texts makes the feature space deviate little from the original. In the parameter space, WiSE-FT [58] proposes interpolating the initial and fine-tuned model for better performance. Inspired by this, we ensemble the weights throughout continuous training to prevent a significant shift from the initial CLIP, which can be seen as interpolating models of different zero-shot transfer and downstream task performance tradeoffs. The weight ensemble method is more stable and not sensitive to hyper-parameters.

To better evaluate our method, we propose a new benchmark Multi-domain Task Incremental Learning (MTIL). Previous CL tasks are crafted by separating classes in one dataset [14, 60, 65], where the images and classes are within a single domain. In contrast, MTIL consists of data from different sources requiring different expert knowledge. It comprises 11 tasks ranging from animal species to aircraft series recognition. As displayed in Fig. 1 (b), when sequentially training CLIP on 11 datasets, the drop in the performance of task i after training task i is the traditional forgetting phenomenon. The degradation in the accuracy compared to the original zero-shot one before training task i represents the forgetting in zero-shot transfer ability. Our method better protects the zero-shot transfer ability and preserves the learned knowledge. We outperform previous methods in both conventional class-incremental learning and MTIL settings. In Fig. 1 (b),

To summarize, our contributions are as follows:

- We investigate continual training with the vision-language model and demonstrate the importance of preserving zero-shot transfer ability. A more challenging benchmark MTIL is proposed to evaluate CL methods where the tasks come from distinct domains.
- We propose a novel method ZSCL to mitigate the catastrophic forgetting problem in continual learning of the vision-language model by distillation in the feature space and weight ensemble in the parameter space.
- The proposed ZSCL outperforms all state-of-the-art methods across multiple benchmark datasets. On 10 steps CL of CIFAR100 and TinyImageNet, our method outperforms the best of previous ones by 7.7% and 6.0% for the Last accuracy. On MTCL, ZSCL outperforms others by 10.9% on Transfer and 9.7% on Avg. scores.

2. Related Work

Vision-Language Models. Inspired by the success of language foundation models such as GPT-3 [3] and T5 [45], a

series of work pre-train vision-language models on large-scale image-text datasets [31, 44, 23]. Among them, Contrastive Language-Vision Pre-training [44] achieves remarkable performance on various downstream tasks. It concentrates on aligning images and texts to acquire a joint embedding space. The CLIP model contains an image encoder [16, 12] and a text encoder [9]. During pre-training, contrastive learning is performed in which a paired image-text is a positive pair while image and text from different image-text pairs form a negative pair. For inference, the closest text embedding for the image is chosen as the prediction. Vision-language models can give zero-shot predictions on unseen tasks with a robust zero-shot transfer ability on various downstream tasks.

Continual Learning Methods. Most existing continual learning methods can be categorized into four groups: parameter expansion, memory replay, distillation loss, and parameter regularization. Parameter expansion methods such as DyTox [14] and DEN [61] introduce new parameters for new tasks. As we want to achieve a more powerful CLIP model at the end of CL, we do not change the architecture of the CLIP model. Memory replay methods [50, 36, 42, 28, 40] including iCaRL [47] and SER [21] keep a memory for exemplars from previously learned tasks. However, pre-training datasets are too large for choosing exemplars or may not be available at downstream training, and downstream data are not good exemplars for preserving the pre-training knowledge. Distillation loss such as LwF [33], LwM [10], LwF-VR [11], and PodNet [13] aligns current output space with previous ones, whereas distillation based on current tasks are not strong enough to maintain foundational knowledge. For the CLIP model, we find that general images, even if never seen by the model, plus sentences with enough semantics, can be a good "replay" for the distillation loss. The parameter regularization loss restricts the flexibility of model parameters by training loss [25, 62, 1] or weight averaging [30, 58]. Although this type of strategy performs badly compared to other ways in previous research [55, 2], we found it helpful for CLIP continual learning. The limited parameter space prevents the model from diverging significantly from its original state.

Vision-Language Models for Downstream Tasks. Many works propose different training strategies of vision-language models for better performance on downstream tasks, such as CoOp [64], CLIP-Adapter [15] and WiSE-FT [58]. However, very few attempts at continual learning exist. While [52, 56] focus on CL in the pretrain of VL and [39] trains VL from scratch using a small dataset, our problem setting differs as we address CL in downstream tasks with a pretrained VL. Currently, most VL models are trained directly on an accessible large dataset, yet integrating knowledge continuously into a pretrained VL is a practical necessity. None of these studies address the zero-shot transfer degradation phenomenon, instead focusing on the traditional forgetting of learned knowledge. Recently, Thengane *et al*. [53] shows CLIP zero-shot prediction achieves state-of-the-art performance in CL settings even without any training. LwF-VR [11] is a modified LwF method for the CLIP model where random-generated sentences are used for distillation loss. However, it only considers the feature space, and the distillation with random sentences cannot protect the vision backbone. Differently, we re-examine what should be used for distillation in the feature space and combine the parameter space weight ensemble to provide better performance for the vision-language model continual learning.

3. Approach

3.1. Preliminaries

Continual Learning. Given n tasks $[\mathcal{T}^1, \mathcal{T}^2, \cdots, \mathcal{T}^n]$, continual training is conducted in sequence on each task $\mathcal{T}^i = (\mathcal{D}^i, C^i), i = 1, \ldots, n$. Here, $\mathcal{D}^i$ represents the task dataset $\{(\boldsymbol{x}_j^i, \boldsymbol{y}_j^i\}_{j=1}^{N_i}$, where $\boldsymbol{x}_j^i$ is an image, $\boldsymbol{y}_j^i$ is a one-hot vector indicating the ground truth, and N_i is the number of images in the dataset. Class names $C^i = \{c_j^i\}_{j=1}^{m_i}$ maps the label of an image to an object name, where m_i is the number of classes for task $\mathcal{T}^i$. The objective of continual learning is to achieve good performance on all tasks.

Two continual learning settings are focused on in this study [55]. In task-incremental learning, at inference, the image $\boldsymbol{x}$ to be predicted is given with its task identity t, so the model only needs to distinguish between different classes in C^t. In class-incremental learning, the task identity t is not given. Thus we need to predict with the combined class set $C = \bigcup_{i=1}^{n} C^i$.

CLIP model. The CLIP model contains an image encoder f_i and a text encoder f_t. The inference process of the CLIP model for image classification is as follows. First, for task $\mathcal{T}^i$, each class c is transformed into a sentence by a template like "a photo of $\{c\}$". Then f_t encodes the classes into text embeddings $\{\boldsymbol{t}_j^i\}_{j=1}^{m_i}$. An image encoder encodes input image $\boldsymbol{x}_k$. The similarity score between the image embedding and text embeddings are calculated as $\boldsymbol{s}_{k,j}^i = \langle f_i(\boldsymbol{x}_k), \boldsymbol{t}_j^i \rangle$, where $\langle \cdot, \cdot \rangle$ denotes the cosine similarity. The class with the largest similarity score is the prediction for the image.

To fine-tune the CLIP model for downstream tasks, cross-entropy loss CE is applied to the similarity score with a temperature scaling:

$$\mathcal{L}_{\text{CE}} = \frac{1}{N} \sum_{i=1}^{N} \text{CE}(\tau \cdot \boldsymbol{s}_i, \boldsymbol{y}_i), \tag{1}$$

where τ is a parameter learned during the pre-training.

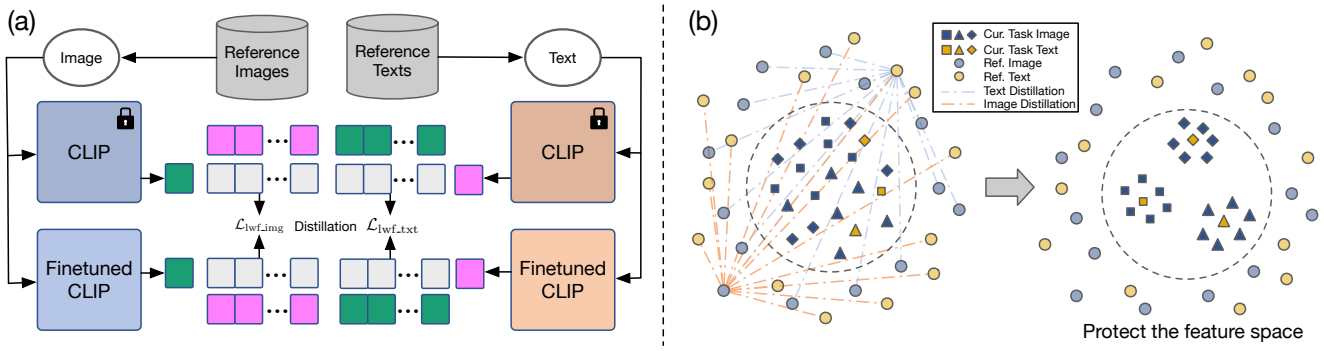


Figure 2. Illustration of ZSCL in feature space. Fig. 3(a) shows the pipeline of distillation. The original and the current model encode the reference images and texts, respectively. The probability distribution of images and texts with respect to each other is distilled. Fig. 3(b) displays how distillation loss preserves the feature space. Compared with the reference dataset, the features of fine-tuning tasks lie in a small subspace. The distillation loss preserves the structure of the feature space by maintaining relative distances.

3.2. Distillation in Feature Space

Well-learned feature space for aligned images and texts enables vision-language models' strong zero-shot transfer ability. It also facilitates the learning of downstream tasks. Compared with the pre-training dataset, downstream datasets lie in a small scope in the feature space (shown in Fig. 2(b)). Direct fine-tuning of downstream tasks greatly distorts the feature distribution of out-of-distribution data, which leads to a significant drop in zero-shot prediction performance.

While the cross-entropy improves the performance by altering fine-tuned feature subspace, we need a new regularization to preserve the potential out-of-distribution feature space. The relative similarity between one image and different texts is:

$$\boldsymbol{p} = \text{Softmax}(\boldsymbol{s}_1, \cdots, \boldsymbol{s}_m). \tag{2}$$

We hope the above similarity distribution is stable during fine-tuning for all potential images and texts. Given a teacher model $\overline{f}$, distillation loss can be applied to penalize changes from the original distribution:

$$\mathcal{L}_{\text{dist\_img}} = \text{CE}(\boldsymbol{p}, \overline{\boldsymbol{p}}) = -\sum_{j=1}^{m} \boldsymbol{p}_j \cdot \log \overline{\boldsymbol{p}}_j, \tag{3}$$

where $\overline{\boldsymbol{p}}$ is the distribution given by the teacher model.

Although the distillation form has been widely used in previous methods [18, 33, 47], they are applied to continual learning from scratch. We investigate different components of the distillation loss for enhancing pre-trained vision-language models.

Three components are discussed in this paper in detail: the data source, the teacher model, and the loss design. Sec. 5.1 shows the performance for the different choices. First, for the data source, LwF [33] uses data from the current task, while iCarl [47] carefully selects data from previous tasks. However, data from downstream tasks span a small subspace and are not good enough to preserve the whole feature space. An ideal choice is the pre-training dataset. However, the CLIP pre-training dataset is private, and the size is too large to load. To solve this challenge, we introduce the reference dataset. A reference dataset is a publicly available image dataset with enough semantics. Enough semantics can be seen as random sampling in the whole feature space. The texts can be related class names, unrelated sentences, or even random tokens. This is because text semantics are easier to sample, and we need not ground truth to calculate the distance between one image to sufficient text samples spread among feature space.

For the teacher model, [33, 47] use the model after learning task i-1 and before learning task i as the teacher model. During the continual training of the vision-language model, the feature space deviates gradually from the initial model. Using fine-tuned models as teacher models enlarges this change. In contrast, we find using the pre-trained model as a teacher model not only preserves the zero-shot transfer ability but also takes advantage of well-learned feature space for better downstream performance.

Finally, previous distillation loss is applied on traditional backbones, where a classification head gives the probability for different labels. For the vision-language model, the probability is calculated based on the relative distance between images and texts. Thus, in addition to Eq. (3), we impose regularization $\mathcal{L}_{\text{dist\_txt}}$ on the distances from a text to a batch of images. The whole framework is shown in Fig. 2 (a) with the following training loss:

$$\mathcal{L} = \mathcal{L}_{\text{ce}} + \lambda \cdot (\mathcal{L}_{\text{lwf\_img}} + \mathcal{L}_{\text{lwf\_txt}}). \tag{4}$$

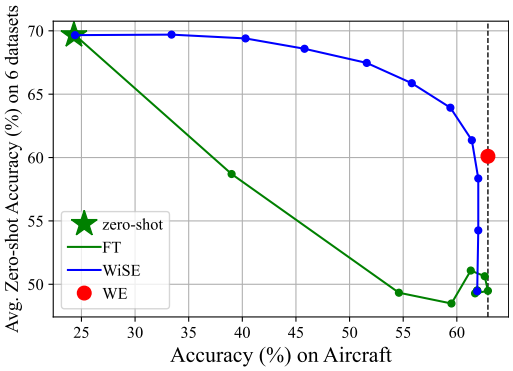


Figure 3. Models during training contain different tradeoffs between zero-shot and new task performance. Points for FT are sampled every 100 iterations, and the ones for WiSE-FT means different α choices. WE ensembles models during training and achieve better performance.

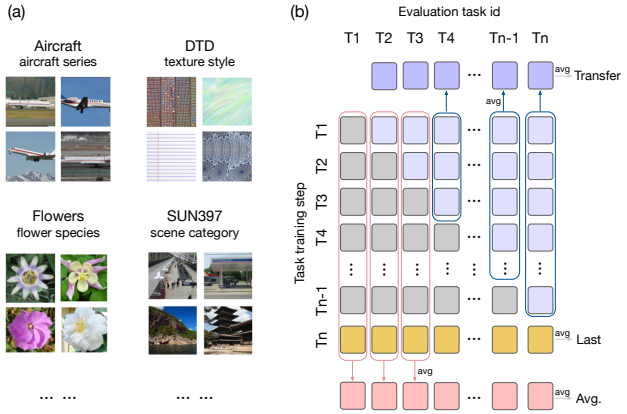


Figure 4. Fig.**(a)**: examples of tasks from different domains in MTIL benchmark. Fig.**(b)**: illustration of calculating metrics Transfer, Avg. and Last during continual learning.

3.3. Weight Ensemble in Parameter Space

Machine learning models integrate learned knowledge in their parameters. To mitigate the forgetting problem, a series of works [25, 62, 1] impose regularization losses on the changes of parameters. Weight consolidation (WC) [25] imposes the following loss:

$$\mathcal{L}_{\text{WC}} = \sum_i (\theta_i - \overline{\theta}_i)^2. \tag{5}$$

where θ is the parameters of the current model, and $\overline{\theta}$ is the reference ones. Although this regularization prevents forgetting, it hinders learning new tasks in a challenging CL setting.

Apart from regularization losses, another method in parameter space is to ensemble different model weights. Model soup [57] averages weights of multiple fine-tuned models to improve the model's robustness but introduces additional training costs. WiSE-FT [58] propose a weighted average between fine-tuned model and the original model to improve the out-of-distribution prediction robustness:

$$f(x; (1-\alpha)\cdot\theta_0 + \alpha\cdot\theta_1), \tag{6}$$

where θ_0 is the original model and θ_1 is the fine-tuned one. However, this method is hyper-parameter-sensitive where different α gives different tradeoffs between zero-shot transfer ability and downstream task performance (blue line in Fig. 3.

Inspired by this, we extend the weighted average to the CL setting. The motivation for the weighted average is to prevent fine-tuning from losing too much knowledge in the original model. As training goes by (green line in Fig. 3), the model performs better on new tasks while losing accuracy on previous ones. The models among training compose a sequence of different learning-forgetting tradeoffs. Instead of interpolating only between the initial and the final model, our method weight ensemble (WE) averages the weights in the sequence during the training time:

$$\hat{\theta}_t = \begin{cases} \theta_0 & t = 0 \\ \frac{1}{t+1}\theta_t + \frac{t}{t+1}\cdot\hat{\theta}_{t-1} & \text{every I iterations} \end{cases}. \tag{7}$$

where model weight sampling happens every I iteration. The method is related to Stochastic Weight Averaging (SWA) [22], but we do not use a modified learning rate schedule here because instead of getting better generalization ability, WE aim to give an improved learning-forgetting tradeoff. As shown in Fig. 3, WE achieves better performance on downstream tasks than WiSE-FT. In addition, while WiSE-FT is sensitive to different values of α, our method is much more robust under different hyper-parameter (I) choices.

4. Multi-domain Task Incremental Learning

Conventional Continual Learning Benchmark. A benchmark consisting of several tasks is needed to evaluate different methods for continual learning. Most previous benchmarks are built by separating classes in a single dataset, such as MNIST [55], CIFAR100 [14], TinyImageNet [60], and ImageNet [60, 65]. We also evaluate our method with traditional benchmarks. In CIFAR100 [26], classes are separated into groups to build tasks. Suppose the dataset has m classes, a k-step setting means we learn m/k classes in each new task. The CIFAR100 contains 100 classes, and the setting of 10, 20, and 50 steps are visited. For TinyImageNet with $m = 200$, the first step learns 100 classes, and the rest is learned with 5, 10, and 20 steps. As for ImageNet-100, we have two settings: ImageNet-100-B0, which includes the same amount of classes for each step, and ImageNet-100-B50, which has

Table 1. **Ablation experiments**. Default settings are marked in gray, which uses image and text distillation loss with the initial CLIP model on 100k ImageNet images and texts generated from ImageNet classes with a simple template.

(a) **Continual learning loss**.

loss	Transfer	Avg.	Last
CE only	44.6	55.9	77.3
Feat. Dist.	47.6	58.7	77.1
Image-only	56.5	68.9	82.1
Text-only	56.7	69.0	82.6
Both	**56.8**	**69.2**	**83.0**

(b) **Data sources for replay**.

source	Transfer	Avg.	Last
current	56.7	66.5	80.2
ImageNet	56.8	**69.2**	**83.0**
CC	**57.2**	68.5	80.9
CIFAR100	55.2	65.9	80.7
Flowers	54.7	66.0	80.8

(c) **Text sources for replay**.

source	Transfer	Avg.	Last
current	51.8	64.9	82.0
prev. all	54.0	70.2	83.7
1k classes (IN)	56.8	69.2	83.0
13k Sent. (CC)	**58.9**	**70.5**	**84.0**
1k Rand. Sent.	58.7	70.2	83.8

(d) **Teacher model**.

source	Transfer	Avg.	Last
Initial	**56.8**	**69.2**	**83.0**
$n-1$	53.9	66.6	80.7
WiSE(0.5)	56.4	68.9	82.9
WiSE(0.8)	56.2	67.8	81.3

(e) **# images for replay**.

#image	Transfer	Avg.	Last
1M	**58.7**	**70.1**	**83.2**
100k	56.8	69.2	83.0
10k	57.8	68.7	81.2
1k	56.3	67.6	80.8

(f) **# image classes for replay**.

#class	Transfer	Avg.	Last
1000	**56.8**	**67.6**	**83.0**
100	56.7	67.3	82.3
10	53.8	66.4	81.0
1	53.1	65.5	80.5

Table 2. Ablation study of different components for ZSCL.

Method	Transfer	Δ	Avg.	Δ	Last	Δ
CLIP `ViT-B/16@224px`						
Zero-shot	69.4	0.0	65.3	0.0	65.3	0.0
Continual Learning	44.6	-24.8	55.9	-9.4	77.3	+12.0
+ Distillation	58.9	-10.5	70.5	+5.2	83.8	+18.5
+ WiSE-FT (best α)	61.7	-7.7	71.6	+6.3	83.3	+18.0
+ WE (ZSCL*)	62.2	-7.2	72.6	+7.3	**84.5**	**+19.2**
+ WC	67.6	-1.8	74.5	+9.2	83.2	+17.9
+ WiSE-FT	67.7	-1.7	74.2	+8.9	81.9	+16.6
+ WE (ZSCL)	**68.1**	**-1.3**	**75.4**	**+10.1**	83.6	+18.3

50 classes for the first step, and the remaining 50 classes are observed progressively over the next 10 stages. For ImageNet [7], we investigate a 10-step setting, which learns 100 new classes per task.

MTIL Benchmark. Different classes from one dataset share the common image source and a similar style [48, 17]. Thus, we propose Multi-domain Task Incremental Learning (MTIL), a cross-domain version of task incremental learning. Different tasks are collected from different domains, requiring different domain knowledge for humans to achieve high accuracy. Our MTIL benchmark consists of 11 tasks (detailed in supplementary materials), as some of the tasks illustrated in Fig. 4 (a). The MTIL benchmark is very challenging with a total number of 1,201 classes. Two orders are used for the evaluation; the first one is alphabet order (Order-I): Aircraft, Caltech101, CIFAR100, DTD, EuroSAT, Flowers, Food, MNIST, OxfordPet, StanfordCars, SUN397. And the second one is a random order (Order-II): StanfordCars, Food, MNIST, OxfordPet, Flowers, SUN397, Aircraft, Caltech101, DTD, EuroSAT, CIFAR100. Experiments are done in Order I by default.

Evaluation Metrics. The metrics of MTIL are illustrated in Fig. 4(b), where rows represent training steps and a column shows performance for one dataset. For conventional continual learning, only scores under the diagonal are meaningful, since they cannot give zero-shot predictions on unseen tasks. In contrast, the zeros-hot transfer ability enables a vision-language model to provide predictions for all datasets. The average accuracy on all datasets among all timestamps is the "Avg" metric. The "Last" metric is the average performance of all tasks after CL. The "Transfer" metric is the average task performance in the upper-right triangle of the matrix. Every task's performance is first averaged to equal the weight of each dataset. It measures to what extent the zero-shot transfer ability is preserved. Before learning task i, tasks not earlier than i are not fine-tuned. Thus, their performance is an indicator of zero-shot transfer ability.

5. Experiments

Implementation. We use CLIP [44] model with image encoder ViT-B/16 [12]. We conduct training with AdamW [37] optimizer and use a label smoothing [38] technique for a better baseline result. For multi-domain task continual learning, we train 1K iterations for each task, while for class-incremental learning, we follow the same evaluation setting in [14]. More implementation details can be found in the supplementary material.

5.1. Main Properties

We ablate our method in feature-space in Tab. 1, and different choices for parameter-space regularization in Tab. 2. Several interesting characteristics are noted.

Continual learning loss. In Tab. 1a, several types of loss for feature space are tested. The Feature Distance penalizing absolute distances achieves a low accuracy. Distillation loss on probability distribution regularizes relative distance in the feature space. With image-only or text-only distilla-

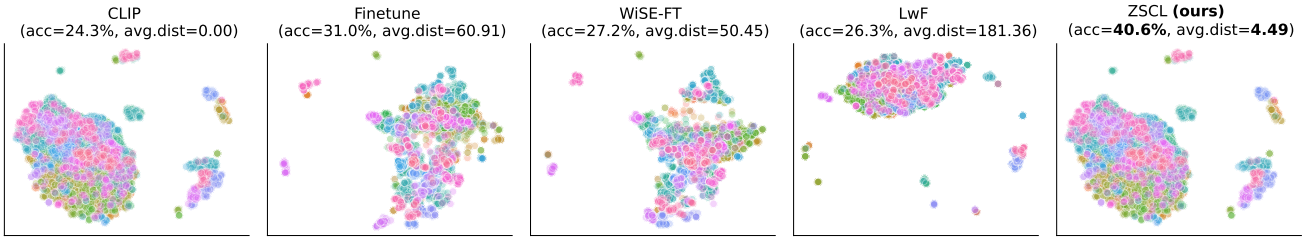


Figure 5. t-SNE on five models' outputs together of Aircraft datasets after MTCL training: only our model maintains a similar feature distribution to the original CLIP ones with minor shift, while the rest significantly distort the feature space.

Table 3. Comparison of different methods on MTIL in Order I.

Method	Transfer	Δ	Avg.	Δ	Last	Δ
CLIP `ViT-B/16@224px`						
Zero-shot	69.4	0.0	65.3	0.0	65.3	0.0
Continual Learning	44.6	-24.8	55.9	-9.4	77.3	+12.0
LwF [33]	56.9	-12.5	64.7	-0.6	74.6	+9.0
iCaRL [47]	50.4	-19.0	65.7	+0.4	80.1	+14.8
LwF-VR [11]	57.2	-12.2	65.1	-0.2	76.6	+11.3
WiSE-FT [58]	52.3	-17.1	60.7	-4.6	77.7	+12.4
ZSCL* (Ours)	62.2	-7.2	72.6	+7.3	**84.5**	**+19.2**
ZSCL (Ours)	**68.1**	**-1.3**	**75.4**	**+10.1**	83.6	+18.3

Table 4. Comparison of different methods on MTIL in Order II.

Method	Transfer	Δ	Avg.	Δ	Last	Δ
CLIP `ViT-B/16@224px`						
Zero-shot	65.4	0.0	65.3	0.0	65.3	0.0
Continual Learning	46.6	-18.8	56.2	-9.1	67.4	+2.1
LwF [33]	53.2	-12.2	62.2	-5.2	71.9	+6.6
iCaRL [47]	50.9	-14.5	56.9	-8.4	71.6	+6.3
LwF-VR [11]	53.1	-12.3	60.6	-7.4	68.3	+0.9
WiSE-FT [58]	51.0	-14.4	61.5	-5.9	72.2	+6.9
ZSCL*	59.8	-5.6	71.8	+6.5	83.3	+18.0
ZSCL	**64.2**	**-1.2**	**74.5**	**+9.2**	**83.4**	**+18.1**

tion, the Transfer, Avg., and Last accuracy all improve. A further boost in performance occurs with both of the distillation losses.

Data source for replay. Tabs. 1b, 1c, 1e and 1f seek the standard for a good reference dataset. As shown in Tab. 1b, distillation on images of current tasks achieves a good Transfer score. However, it hinders the learning on new tasks and results in a low Avg. and Last score. Images in a specific domain (*e.g.*, Flowers) are also not good choices. General images in ImageNet and Conceptual Caption (CC) datasets are examples of good reference datasets. These images are easily available by a web crawler [54]. Text with more semantics can improve performance (shown in Tab. 1c). When using sentences from the Conceptual Caption dataset [49], or even sentences randomly generated from the CLIP vocabulary, there is no ground truth target among the texts for the image from the ImageNet dataset. However, they all achieve an improvement due to more semantics. The reference image dataset does not need to be labeled, matched with the text, or seen by the CLIP model.

Enough semantics in the reference image dataset boosts the distillation performance. In Tabs. 1e and 1f, a smaller number of images and classes all lead to a degradation in the performance. Fewer classes of images in the reference dataset have a worse impact on the performance compared with the image numbers. To keep a reasonable memory buffer, we randomly sample 100k images from ImageNet for MTIL and use texts from CC dataset. For class-incremental learning, conceptual caption dataset's validation set (28k images) is used to avoid information leakage.

Teacher model. Tab. 1d shows the performance of distillation loss with different teacher models. Unlike conventional continual learning, the teacher model should not be the one trained on the previous task; otherwise, the deviation from the initial CLIP in the previous task may be amplified. In contrast, with the initial CLIP as the teacher model, not only is the zero-shot performance improved but the Mean and Last scores are also boosted, indicating that preserving high-quality feature space facilitates continual learning.

Parameter-space regularization. In Tab. 2, we experiment with three different parameter-space regularizations. We experiment with two variants of WiSE-FT [58]: the weighted average between the current model with the initial CLIP or the one at the previous task. The result shows the latter one is a better choice because keeping weight averaging with initial CLIP loses the newly learned knowledge. We experiment with different α choices for WiSE, and the best result is reported. While distillation loss improves the whole performance, the parameter-space regularization further protects the zero-shot transfer ability with a higher Transfer. Among the three parameter-space regularizations, only WE achieve a better Last score. WC greatly improves the Transfer scores with a lower Last score. A combination of the weight consolidation loss with weight

Table 5. Transfer, Avg., and Last scores (%) of different continue training methods on MTIL benchmark.

Method	Aircraft	Caltech101	CIFAR100	DTD	EuroSAT	Flowers	Food	MNIST	OxfordPet	Cars	SUN397
CLIP `ViT-B/16@224px`											
Zero-shot	24.3	88.4	68.2	44.6	54.9	71.0	88.5	59.4	89.0	64.7	65.2
Fine-tune	62.0	95.1	89.6	79.5	98.9	97.5	92.7	99.6	94.7	89.6	81.8
Transfer											
Continual-FT		67.1	46.0	32.1	35.6	35.0	57.7	44.1	60.8	20.5	46.6
LwF [33]		74.5	56.9	39.1	51.1	52.6	72.8	60.6	75.1	30.3	55.9
iCaRL [47]		56.6	44.6	32.7	39.3	46.6	68.0	46.0	77.4	31.9	60.5
LwF-VR [11]		77.1	61.0	40.5	45.3	54.4	74.6	47.9	76.7	36.3	58.6
WiSE-FT [58]		73.5	55.6	35.6	41.5	47.0	68.3	53.9	69.3	26.8	51.9
Dist. only		80.1	62.2	40.2	39.9	58.1	80.8	53.4	74.6	38.1	61.9
ZSCL*		78.3	64.0	42.9	45.2	63.5	84.2	56.1	78.9	44.1	64.3
ZSCL		**86.0**	**67.4**	**45.4**	**50.4**	**69.1**	**87.6**	**61.8**	**86.8**	**60.1**	**66.8**
Avg.											
Continual-FT	25.5	81.5	59.1	53.2	64.7	51.8	63.2	64.3	69.7	31.8	49.7
LwF [33]	36.3	86.9	72.0	59.0	73.7	60.0	73.6	74.8	80.0	37.3	58.1
iCaRL [47]	35.5	89.2	72.2	60.6	68.8	70.0	78.2	62.3	81.8	41.2	62.5
LwF-VR [11]	29.6	87.7	74.4	59.5	72.4	63.6	77.0	66.7	81.2	43.7	60.7
WiSE-FT [58]	26.7	86.5	64.3	57.1	65.7	58.7	71.1	70.5	75.8	36.9	54.6
Dist. only	48.1	90.6	79.8	63.2	75.6	72.5	84.7	70.2	79.8	46.9	63.7
ZSCL*	**50.7**	90.9	79.8	63.8	76.6	77.3	87.0	71.9	83.0	52.0	65.9
ZSCL	45.1	**92.0**	**80.1**	**64.3**	**79.5**	**81.6**	**89.6**	**75.2**	**88.9**	**64.7**	**68.0**
Last											
Continual-FT	31.0	89.3	65.8	67.3	88.9	71.1	85.6	99.6	92.9	77.3	81.1
LwF [33]	26.3	87.5	71.9	66.6	79.9	66.9	83.8	99.6	92.1	66.1	80.4
iCaRL [47]	35.8	**93.0**	77.0	70.2	83.3	88.5	90.4	86.7	93.2	81.2	**81.9**
LwF-VR [11]	20.5	89.8	72.3	67.6	85.5	73.8	85.7	99.6	93.1	73.3	80.9
WiSE-FT [58]	27.2	90.8	68.0	68.9	86.9	74.0	87.6	99.6	92.6	77.8	81.3
Dist. only	43.3	91.9	**81.3**	**72.4**	**95.1**	90.5	90.4	**99.7**	92.5	85.1	81.8
ZSCL*	**46.0**	92.3	81.2	**72.4**	93.0	**92.1**	90.8	99.6	93.3	**86.6**	81.7
ZSCL	40.6	92.2	**81.3**	70.5	94.8	90.5	**91.9**	98.7	**93.9**	85.3	80.2

ensemble achieves a better tradeoff between Transfer and Last value. While ZSCL*, a variant without the WC loss, obtains the highest Last score, the ZSCL with WC loss outperforms it with 2.8% Avg. and 5.9% Transfer scores.

5.2. Multi-domain Task Incremental Learning

Tab. 3 displays the performance of different methods on the MTIL benchmark, and Tab. 5 presents the detailed Transfer, Avg, and Last metrics on each dataset. Zero-shot denotes the zero-shot prediction performance of the initial CLIP model, and Fine-tune means the direct fine-tuning accuracy on each dataset, which can be seen as an upper-bound where no forgetting phenomenon happens. Continual learning uses cross-entropy loss to learn each dataset sequentially, where there is a significant forgetting issue on both zero-shot predictions (Transfer drops by 24.8%) and newly learned knowledge (Last drops by 9.4%). Previous methods improve the Last performance slightly and cannot maintain a high zero-shot prediction performance. Without WC, ZSCL* achieves the best Last scores, outperforming the previous best one by 4.4%. Our method ZSCL improves 9.1% on Transfer accuracy, with only 1.3% drops compared with the initial CLIP model, and achieves a 10.1% gain in the Avg. accuracy.

Figure Fig. 5 provides a visualization of the feature space on Aircraft of the original CLIP and four methods trained at the end of MTCL (t-SNE conducted only once on all features collected). This shows our method is capable of maintaining the pretrained feature distribution with a small averaged feature distance and thus preserving zero-shot performance.

The result of the MTIL method in Order-II is presented in Tab. 4. Our method surpasses previous methods in another order setting of the MTIL benchmark. A similar conclusion holds from the results of MTIL Order-II compared with MTIL Order-I. Our method ZSCL outperforms others by 9.2% on the Avg. score and 18.1% on the Last score with only a 1.2% performance loss on the Transfer score. This shows our approach can work for different orders of the multi-domain task incremental learning. In addition,

Table 6. Comparison of state-of-the-art CL methods on CIFAR100 benchmark in class-incremental setting.

	10 steps		20 steps		50 steps	
Methods	**Avg**	**Last**	**Avg**	**Last**	**Avg**	**Last**
UCIR [19]	58.66	43.39	58.17	40.63	56.86	37.09
BiC [59]	68.80	53.54	66.48	47.02	62.09	41.04
RPSNet [46]	68.60	57.05	-	-	-	-
PODNet [13]	58.03	41.05	53.97	35.02	51.19	32.99
DER [60]	74.64	64.35	73.98	62.55	72.05	59.76
DyTox+ [14]	74.10	62.34	71.62	57.43	68.90	51.09
CLIP [44]	74.47	65.92	75.20	65.74	75.67	65.94
FT	65.46	53.23	59.69	43.13	39.23	18.89
LwF [33]	65.86	48.04	60.64	40.56	47.69	32.90
iCaRL [47]	79.35	70.97	73.32	64.55	71.28	59.07
LwF-VR [11]	78.81	70.75	74.54	63.54	71.02	59.45
ZSCL (Ours)	**82.15**	**73.65**	**80.39**	**69.58**	**79.92**	**67.36**
Impr	**+7.68**	**+7.73**	**+5.19**	**+3.84**	**+3.95**	**+1.42**

Table 7. Comparison of different methods on TinyImageNet splits in class-incremental settings with 100 base classes.

	5 steps		10 steps		20 steps	
Methods	**Avg**	**Last**	**Avg**	**Last**	**Avg**	**Last**
EWC [25]	19.01	6.00	15.82	3.79	12.35	4.73
EEIL [4]	47.17	35.12	45.03	34.64	40.41	29.72
UCIR [19]	50.30	39.42	48.58	37.29	42.84	30.85
MUC [35]	32.23	19.20	26.67	15.33	21.89	10.32
PASS [65]	49.54	41.64	47.19	39.27	42.01	32.93
DyTox [14]	55.58	47.23	52.26	42.79	46.18	36.21
CLIP [44]	69.62	65.30	69.55	65.59	69.49	65.30
FT	61.54	46.66	57.05	41.54	54.62	44.55
LwF [33]	60.97	48.77	57.60	44.00	54.79	42.26
iCaRL [47]	77.02	70.39	73.48	65.97	69.65	64.68
LwF-VR [11]	77.56	70.89	74.12	67.05	69.94	63.89
ZSCL (Ours)	**80.27**	**73.57**	**78.61**	**71.62**	**77.18**	**68.30**
Impr	**+10.65**	**+8.27**	**+9.06**	**+6.03**	**+7.69**	**+3.00**

compared with Order-I, previous methods achieve a much lower Last score (e.g., for Continual-Learning, Order-I has 77.3%, while Order-II has 65.3%). With ZSCL, the Last score is similar (83.6% compared with 83.4%). This shows our method is more robust towards different training orders.

5.3. Class Incremental Learning

We evaluate our methods on conventional class incremental learning. Tabs. 6 and 7 display results on CIFAR100 and TinyImageNet, respectively. We re-implement some previous methods with a CLIP backbone (after CLIP in the table), while others using a special network design cannot be easily adapted. Although zero-shot CLIP prediction achieves a good result on these benchmarks, continual learning with direct fine-tuning or LwF [33] degrades the performance greatly, especially under the setting of a large step number. This demonstrates a severe catastrophic forgetting phenomenon in fine-tuning the CLIP model. Our method consistently improves the performance of the CLIP model on both Avg. and Last scores with a large gap towards previous ones.

6. Limitation and Future Work

Our proposed method has a limitation that a reference dataset is needed. A promising direction of the work is to preserve the zero-shot transfer ability without the need for an outside dataset. For example, we may generate a synthetic image dataset as the reference dataset. Methods like [51] can synthesize datasets from a network.

The deep learning community tends to build large models with a huge dataset [3, 6], including vision-language models [44, 20]. As re-training cost upsurges, continual learning is an efficient approach for updating these models with custom usage.

In some cases, we want to correct wrong information in the pre-training dataset or update old information with the latest one. How to conduct this task with a reference dataset is left as future work.

Lately, a noticeable trend involves the creation of multi-modality models utilizing large language models, showcasing encouraging outcomes in tasks such as Visual Question Answering (VQA). Expanding our approach to encompass the next token prediction task remains an avenue for future exploration and research.

7. Conclusion

In this paper, we investigate continual learning with the vision-language model. We propose a better continual learning algorithm to protect the zero-shot transfer ability in the vision-language model learned in the pre-training stage. Our algorithm mitigates the catastrophic forgetting in both feature space and parameter space. In feature space, distilling the initial model on a reference dataset significantly boosts the model's performance. In parameter space, weight ensemble among different training stages alleviates the forgetting issue. We propose a more challenging Multi-domain Task Incremental Learning (MTIL) benchmark to evaluate the continual learning methods better. On both conventional and new benchmarks, our method achieves state-of-the-art performance.

Acknowledgements

Yang You's research group is being sponsored by the NUS startup grant (Presidential Young Professorship), Singapore MOE Tier-1 grant, ByteDance grant, ARCTIC grant, SMI grant, and Alibaba grant.

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 139–154, 2018. 3, 5

[2] Eden Belouadah, Adrian Popescu, and Ioannis Kanellos. A comprehensive study of class incremental learning algorithms for visual tasks. *Neural Networks*, 135:38–54, 2021. 3

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 2, 9

[4] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 233–248, 2018. 9

[5] Zhou Daquan, Kai Wang, Jianyang Gu, Xiangyu Peng, Dongze Lian, Yifan Zhang, Yang You, and Jiashi Feng. Dataset quantization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 2

[6] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. *arXiv preprint arXiv:2302.05442*, 2023. 9

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6

[8] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012. 2

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[10] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5138–5146, 2019. 2, 3

[11] Yuxuan Ding, Lingqiao Liu, Chunna Tian, Jingyuan Yang, and Haoxuan Ding. Don't stop learning: Towards continual learning for the clip model. *arXiv preprint arXiv:2207.09248*, 2022. 2, 3, 7, 8, 9

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 3, 6

[13] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *European Conference on Computer Vision*, pages 86–102. Springer, 2020. 2, 3, 9

[14] Arthur Douillard, Alexandre Ramé, Guillaume Couairon, and Matthieu Cord. Dytox: Transformers for continual learning with dynamic token expansion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9285–9295, 2022. 2, 3, 5, 6, 9

[15] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021. 3

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 3

[17] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021. 6

[18] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015. 2, 4

[19] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 831–839, 2019. 9

[20] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*, 2023. 9

[21] David Isele and Akansel Cosgun. Selective experience replay for lifelong learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2, 3

[22] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018. 5

[23] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. 2, 3

[24] Woojeong Jin, Yu Cheng, Yelong Shen, Weizhu Chen, and Xiang Ren. A good prompt is worth millions of parameters? low-resource prompt-based learning for vision-language models. *arXiv preprint arXiv:2110.08484*, 2021. 2

[25] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 3, 5, 9

[26] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. 5

[27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 1

[28] Frantzeska Lavda, Jason Ramapuram, Magda Gregorova, and Alexandros Kalousis. Continual classification learning using generative models. *arXiv preprint arXiv:1810.10612*, 2018. 2, 3

[29] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2

[30] Sang-Woo Lee, Jin-Hwa Kim, Jaehyun Jun, Jung-Woo Ha, and Byoung-Tak Zhang. Overcoming catastrophic forgetting by incremental moment matching. *Advances in neural information processing systems*, 30, 2017. 3

[31] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7331–7341, 2021. 3

[32] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, 2022. 2

[33] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 1, 2, 3, 4, 7, 8, 9

[34] Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. Dream: Efficient dataset distillation by representative matching. *ICCV-2023*, 2023. 2

[35] Yu Liu, Sarah Parisot, Gregory Slabaugh, Xu Jia, Ales Leonardis, and Tinne Tuytelaars. More classifiers, less forgetting: A generic multi-classifier paradigm for incremental learning. In *European Conference on Computer Vision*, pages 699–716. Springer, 2020. 9

[36] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017. 2, 3

[37] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[38] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019. 6

[39] Zixuan Ni, Longhui Wei, Siliang Tang, Yueting Zhuang, and Qi Tian. Continual vision-language representaion learning with off-diagonal information. *arXiv preprint arXiv:2305.07437*, 2023. 3

[40] Oleksiy Ostapenko, Timothee Lesort, Pau Rodríguez, Md Rifat Arefin, Arthur Douillard, Irina Rish, and Laurent Charlin. Continual learning with foundation models: An empirical study of latent replay. In *Conference on Lifelong Learning Agents*, pages 60–91. PMLR, 2022. 3

[41] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3498–3505, 2012. 1

[42] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *European conference on computer vision*, pages 524–540. Springer, 2020. 2, 3

[43] Ziheng Qin, Kai Wang, Zangwei Zheng, Jianyang Gu, Xiangyu Peng, Daquan Zhou, and Yang You. Infobatch: Lossless training speed up by unbiased dynamic data pruning. *arXiv preprint arXiv:2303.04947*, 2023. 2

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2, 3, 6, 9

[45] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020. 2

[46] Jathushan Rajasegaran, Munawar Hayat, Salman Khan, Fahad Shahbaz Khan, Ling Shao, and Ming-Hsuan Yang. An adaptive random path selection approach for incremental learning. *arXiv preprint arXiv:1906.01120*, 2019. 9

[47] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1, 2, 3, 4, 7, 8, 9

[48] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pages 5389–5400. PMLR, 2019. 6

[49] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 7

[50] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017. 2, 3

[51] James Smith, Yen-Chang Hsu, Jonathan Balloch, Yilin Shen, Hongxia Jin, and Zsolt Kira. Always be dreaming: A new approach for data-free class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9374–9384, 2021. 9

[52] Tejas Srinivasan, Ting-Yun Chang, Leticia Pinto Alva, Georgios Chochlakis, Mohammad Rostami, and Jesse Thomason. Climb: A continual learning benchmark for vision-and-language tasks. *Advances in Neural Information Processing Systems*, 35:29440–29453, 2022. 3

[53] Vishal Thengane, Salman Khan, Munawar Hayat, and Fahad Khan. Clip model is an efficient continual learner. *arXiv preprint arXiv:2210.03114*, 2022. 2, 3

[54] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 7

[55] Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019. 3, 5

[56] Zhen Wang, Liu Liu, Yiqun Duan, Yajing Kong, and Dacheng Tao. Continual learning with lifelong vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 171–181, 2022. 3

[57] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, pages 23965–23998. PMLR, 2022. 5

[58] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7959–7971, 2022. 2, 3, 5, 7, 8

[59] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019. 2, 9

[60] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023, 2021. 2, 5, 9

[61] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. *arXiv preprint arXiv:1708.01547*, 2017. 3

[62] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995. PMLR, 2017. 3, 5

[63] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022. 2

[64] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 2, 3

[65] Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5871–5880, 2021. 2, 5, 9