Continual Learning of Image Classes With Language Guidance From a Vision-Language Model

Wentao Zhang, Yujun Huang, Weizhuo Zhang, Tong Zhang, *Member, IEEE*, Qicheng Lao, Yue Yu, Wei-Shi Zheng, and Ruixuan Wang

***Abstract*— Current deep learning models often catastrophically forget the knowledge of old classes when continually learning new ones. State-of-the-art approaches to continual learning of image classes often require retaining a small subset of old data to partly alleviate the catastrophic forgetting issue, and their performance would be degraded sharply when no old data can be stored due to privacy or safety concerns. In this study, inspired by human learning of visual knowledge with the effective help of language, we propose a novel continual learning framework based on a pre-trained vision-language model (VLM) without retaining any old data. Rich prior knowledge of each new image class is effectively encoded by the frozen text encoder of the VLM, which is then used to guide the learning of new image classes. The output space of the frozen text encoder is unchanged over the whole process of continual learning, through which image representations of different classes become comparable during model inference even when the image classes are learned at different times. Extensive empirical evaluations on multiple image classification datasets under various settings confirm the superior performance of our method over existing ones. The source code is available at https://github.com/Fatflower/CIL_LG_VLM/.**

***Index Terms*— Continual learning, vision-language model, language guidance.**

Manuscript received 30 April 2024; revised 10 July 2024; accepted 13 August 2024. Date of publication 23 August 2024; date of current version 23 December 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62071502, in part by the Major Key Project of Peng Cheng Laboratory (PCL) under Grant PCL2023A09, and in part by Guangdong Excellent Youth Team Program under Grant 2023B1515040025. This article was recommended by Associate Editor Y. S. Rawat. *(Corresponding authors: Yue Yu; Ruixuan Wang.)*

Wentao Zhang, Yujun Huang, Weizhuo Zhang, and Wei-Shi Zheng are with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510275, China, and also with the Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou 510275, China (e-mail: zhangwt65@mail2.sysu.edu.cn; huangyj273@mail2.sysu.edu.cn; zhangwzh26@mail2.sysu.edu.cn; wszheng@ieee.org).

Tong Zhang and Yue Yu are with the Peng Cheng Laboratory, Shenzhen 518066, China (e-mail: zhangt02@pcl.ac.cn; yuy@pcl.ac.cn).

Qicheng Lao is with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: qicheng.lao@gmail.com).

Ruixuan Wang is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510275, China, also with the Peng Cheng Laboratory, Shenzhen 518066, China, and also with the Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou 510275, China (e-mail: wangruix5@mail.sysu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2024.3449109.

Digital Object Identifier 10.1109/TCSVT.2024.3449109

I. Introduction

DEEP learning has achieved remarkable performance in various applications, e.g., image recognition [1], [2], semantic segmentation [3], [4], and natural language processing [5], [6], [7]. However, current intelligent systems lack the continual or lifelong learning ability as human beings. Such ability is crucial when new knowledge needs to be continually learned after the intelligent system is deployed, such as in automated retail stores [8] and learning to diagnose new emerging diseases. When an intelligent system is updated to learn new knowledge with training data of new classes, it often catastrophically forgets previously learned old knowledge [9].

Multiple approaches have been proposed to alleviate such catastrophic forgetting issue. One group of approaches presume that old knowledge is implicitly stored in model parameters, and try to find those model parameters that are crucial for old knowledge and keep them unchanged when the model learns new knowledge [10], [11], [12]. However, due to the nonlinear property of deep learning models and complicated interactions between model parameters across layers, value changes in some parameters would more or less change the memory of old knowledge implicitly stored by the whole model. On the other hand, more and more model parameters become crucial for increasing old knowledge over multiple rounds of continual learning. Keeping more parameters unchanged would make the model difficult to learn new knowledge in subsequent rounds.

To make the model capable of learning new knowledge (aka 'plasticity') and keeping old classes of knowledge (aka 'stability'), researchers found that retaining a small subset of data from each previously learned class is very helpful [13], [14], [15]. With the retained small old data, knowledge distillation techniques can be employed to help update the model such that the updated model has similar output responses at various network layers as the old model, meanwhile, existing model components or only newly added modules can be fine-tuned to learn new knowledge from the training set of new classes. In this way, old knowledge is largely preserved in the updated model by presuming that Model A contains knowledge of Model B if output response of Model A is similar to that of Model B for any model input. However, due to severe imbalance between new classes of data and

accessible old data, the updated model is often biased toward currently learned new classes during inference, suggesting that knowledge distillation with limited old data is not enough to well keep old knowledge. Class-rebalancing strategies during model updating can only slightly alleviate the severe class imbalance issue [16], [17], [18]. Even worse, old data may not be accessible at all in some scenarios due to privacy or safety concerns.

When no old data is available, prompt learning [19] built on certain pre-trained and frozen vision models, particularly the Vision Transformer (ViT) [1], has shown promising continual learning performance [20], [21]. In this approach, only a set of 'prompts' as part of the input to one or more model layers are learned during continual learning. The efficacy of this approach largely depends on the assumption that visual features extracted from one or more layers of the pre-trained model can effectively discriminate between old and new knowledge, which may not be valid especially when visually similar classes are learned at different rounds of continual learning.

Inspired by the observation that humans can effectively learn new visual knowledge with the help of language and by the wide applications of pre-trained vision-language models (VLMs) [22], [23], we propose a novel VLM-based continual learning framework without retaining any old data. In the framework, only the inserted light-weight visual adapters [24] at each round of continual learning are learnable, and the prior knowledge of each new visual class encoded by the frozen text encoder is used to effectively guide the training of visual adapters. Since the semantic textual space (i.e., output space of text encoder) is unchanged over multiple rounds of continual learning, and each class of images is clustered around the associated distinctive and unchanged textual representation of the same class, images from different classes (including old and new classes) become largely differentiated in the semantic textual space. As a result, the updated model can well recognize both old and new classes with the help of the unchanged semantic textual space. In addition, with the help of mixed textual representations from multiple classes, the model can be trained to further improve its continual learning performance and additionally improve the out-of-distribution (OOD) detection performance. Extensive empirical evaluations on three image classification datasets confirm the superior performance of the proposed framework over state-of-the-art approaches which even use retained old data during continual learning (see Figure 1 for an example). The main contributions are summarized below.

- A novel and effective VLM-based continual learning framework. It does not retain any old data and uses the rich textual knowledge of each visual class to guide the training of visual adapters in the fixed textual space.
- First time to show an unchanged semantic textual space together with rich textual descriptions of each class is effective to help continual learning of visual classes.
- First time to show faked textual OOD representations can help improve both continual learning and OOD detection of visual classes.

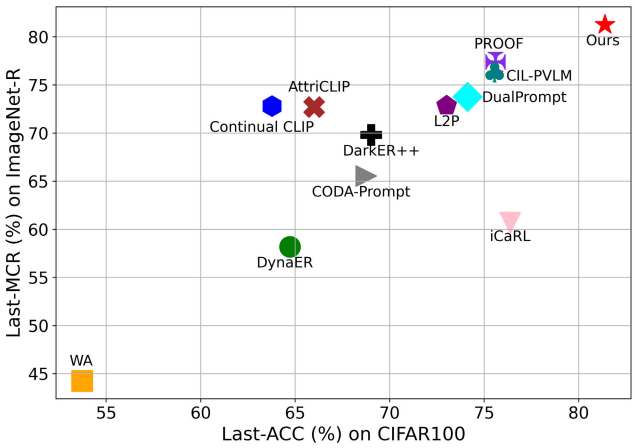


Fig. 1. Continual learning performance from different methods on CIFAR100 (10 classes per round) and ImageNet-R (20 classes per round). 'Last-ACC'/'Last-MCR': accuracy/mean class recall after learning all classes.

- Extensive empirical evaluations on multiple benchmarks, with state-of-the-art performance achieved.

II. Related Work

Continual learning approaches can be grouped into four categories: regularization-based, model expansion-based, knowledge distillation-based, and pre-trained model-based.

Regularization-based methods [5], [10], [12], [25], [26] focus on identifying and minimally altering crucial components of the model for retention of previously learned knowledge while acquiring new knowledge. The importance of model parameters is often assessed based on their sensitivity to changes in the loss function. For example, SOUL [5] designs a local topology preservation loss to prevent the topological relationship of the learned feature space from drifting; spWC [12] prioritizes past tasks based on key performance indicators such as accuracy, ensuring that when learning new tasks, the model selectively retains knowledge from more difficult past tasks. While effective in preserving old knowledge at the initial phases of continual learning, they tend to gradually accumulate a large number of safeguarded parameters, which eventually hinders the learning of new knowledge.

Expansion-based methods learn new knowledge by modifying the structure of the network, e.g., by adding layers, subnetworks, or a new feature extractor [18], [27], [28], [29]. An example is MoBoo [28], which designs a memory-enhanced attention mechanism for new knowledge into an updated classifier. However, such methods often cause model size increased quickly over rounds of continual learning.

In contrast, knowledge distillation-based methods [15], [30], [31], [32] distill old knowledge from old model into the new one, which typically involves using a small number of stored old data. However, their effectiveness decreases as more knowledge is learned, mainly due to insufficient representation of old knowledge by the stored small data.

In addition, with the increasing popularity of visual pre-trained models, methods utilizing pre-trained models have been proposed to preserve old knowledge while learning

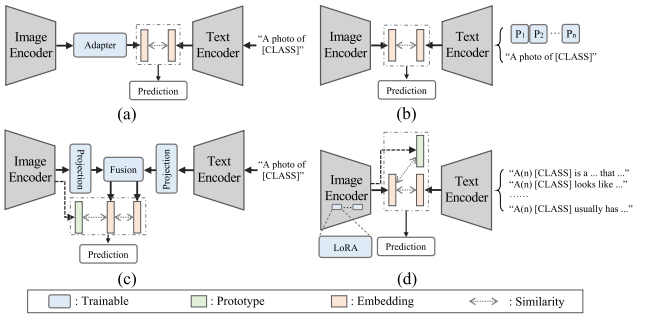


Fig. 2. VLM-based continual learning. (a) CIL-PVLM [33] adds an adapter into the visual part of the VLM. (b) AttriCLIP [34] adds learnable prompts into the textual part of the VLM. (c) PROOF [35] adds task-specific projections into both visual and textual parts of the VLM with a cross-attention module to fuse the visual and textual information. (d) Ours uses the rich textual knowledge of each visual class to guide the training of the image encoder in an unchanged textual space.

new knowledge [20], [21], [36], [37], [38]. They effectively utilize the knowledge of the pre-trained vision models and can achieve promising performance using efficient fine-tuning of the parameters. However, these methods, which mainly rely on visual information for decision-making, often struggle with the misclassification of targets with similar visual features. With recent development of large vision-language models (VLMs) [22], [23], [39], [40], [41], several approaches have applied pre-tained VLMs to continual learning [33], [34], [35]. However, they change the textual output space of the text encoder over continual learning (Figure 2b, 2c), not using existing rich prior knowledge of each visual class (Figure 2a, 2b, 2c), and requiring retained old data when learning new knowledge (Figure 2a, Figure 2c). In contrast, our VLM-based method utilizes rich knowledge of each visual class and the fixed semantic textual space to effectively guide the training the of image classifier.

III. Method

In class-incremental learning (CIL) without retaining old data, the model learns a certain number (c_t) of new classes at the t-th round (also called t-th task) of continual learning based on the training set of the solely c_t new classes, and no data of any previously learned classes are involved. In the t-th round, after finishing learning the c_t new classes from the t-th task, the model is expected to recognize all the $C_t = c_1 + c_2 + \ldots + c_t$ classes learned so far.

A. Framework Overview

We propose a continual learning framework based on a pre-trained vision-language model (VLM) (Figure 3). The core idea is using language to guide the training of the image classifier when the classifier learns to recognize new image classes, motivated by human learning of visual knowledge with the effective help of language. The architecture of the framework mainly consists of two parts, i.e., the vision part (Figure 3, top left) and the language part (Figure 3, top right).

In the vision part, the original pre-tained VLM image encoder is frozen, and only the newly added visual adapters (LoRA in Figure 3, bottom left, in light red) specifically for the c_t new classes are optimized with the guidance from the outputs of the language part (Figure 3, top middle). Since a set of unique visual adapters are optimized for the new classes at each continual learning round, and old knowledge learned at each previous round is perfectly preserved in the corresponding old set of unique visual adapters (together with the frozen VLM's image encoder), the updated image classifier model well learns the visual knowledge of new classes without forgetting any old knowledge. Thus, the image classifier is enabled with both desired plasticity and stability properties during continual learning. Note that the visual adapters contain a relatively small number of parameters compared to the original VLM's image encoder, and therefore the whole model size is increased very slowly over multiple rounds of continual learning.

The language part is frozen and provides distinctive semantic textual representation for each visual class mainly with the help of a large language model (LLM, e.g., ChatGPT [42]) and a pre-trained VLM's text encoder. The semantic textual space (i.e., output space of the VLM's text encoder) is unchanged over the whole process of continual learning and shared by all rounds of continual learning tasks. Therefore, guiding the learning of new visual classes by the distinctive semantic textual representation of each class in the unchanged textual space (see Section III-C for details) would make the visual representations of both old and new classes comparable in the unchanged textual space. In addition, textual representations of multiple classes can be mixed together to generate faked out-of-distribution (OOD) representations which do not belong to any visual classes. Such faked OOD textual representations can be also utilized to help improve the model's performance in both continual learning and OOD detection. Last but not least, with the help of the unchanged textual space, integrating both the textual and visual information can further improve the prediction accuracy during inference.

B. Task-Specific Visual Adapters

Adapter techniques such as delta tuning [43] and LoRA [24] have been recently proposed to fine-tune large language models for various downstream language tasks, and such techniques have also been extended to fine-tune large vision models [44], [45]. The main idea is to insert a new module called adapter into one or more layers of the pre-trained large model, and the original modules of the pre-trained model are frozen and only the new adapter(s) are optimized during model fine-tuning. In our learning framework, when the model learns the c_t new visual classes at the t-th round of continual learning, a set of new adapters specifically for the c_t new classes are inserted into the pre-trained VLM image encoder (so called 'task-specific visual adapters'). Specifically, with the default VLM model CLIP and the ViT backbone for its image encoder, one pair of LoRA adapters are inserted into each self-attention head at each self-attention layer of the image encoder, one for the projection of key tokens and the other for the projection of value tokens (Figure 3, lower left). Each LoRA adapter simply consists of two linear layers (without activation function). Formally, let $\mathbf{A}_{l,h} \in \mathbb{R}^{d \times v}$ denote the learnable LoRA adapter

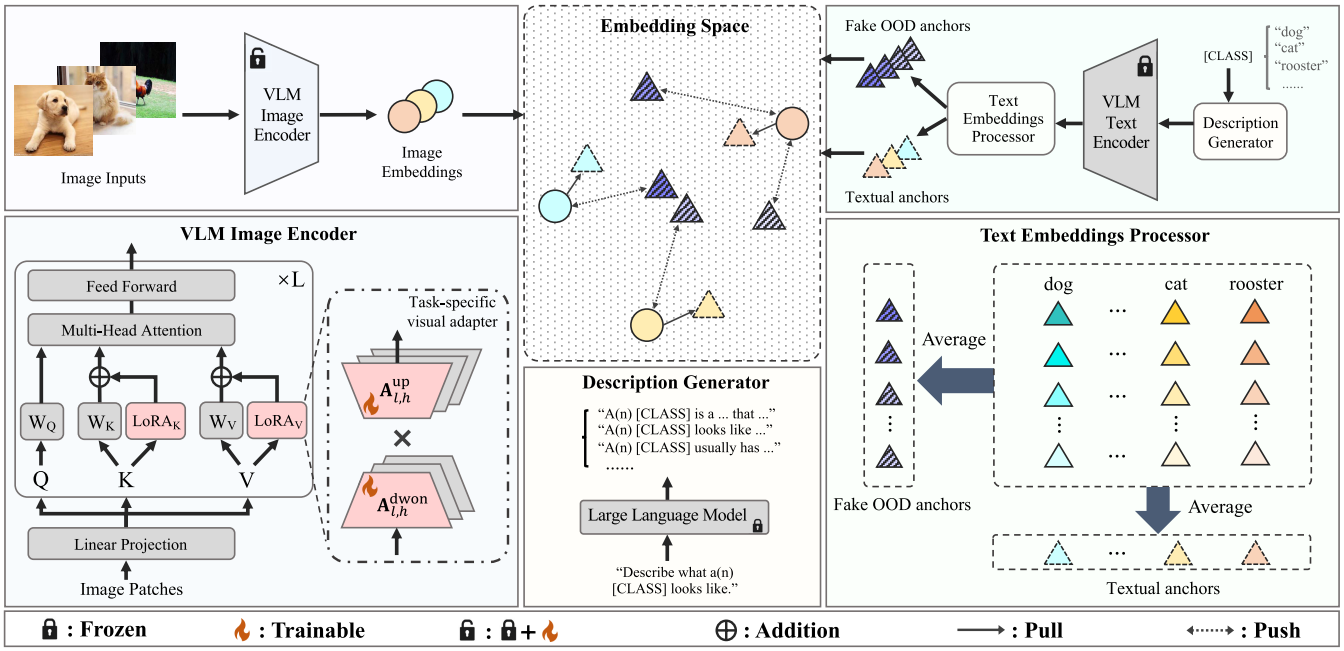


Fig. 3. The proposed VLM-based continual learning framework. A set of new visual adapters (LoRA) specifically for the new set of classes are inserted into the pre-trained VLM image encoder (lower left). It utilizes the powerful textual representation capability of the pre-trained VLM text encoder to guide the training of visual adapters (upper part). Lower middle: acquisition of rich textual descriptions for each class. Lower right: generation of textual anchors and fake OOD anchors. 'L': the number of layers in the VLM image encoder.

for value token projection in the h-th head at the l-th self-attention layer, where v and d are respectively the input and output dimension of the h-th head for value token projection. Then

$$\mathbf{A}_{l,h} = \mathbf{A}^{\text{up}}_{l,h}\mathbf{A}^{\text{down}}_{l,h}, \tag{1}$$

where $\mathbf{A}^{\text{down}}_{l,h} \in \mathbb{R}^{r\times v}$ and $\mathbf{A}^{\text{up}}_{l,h} \in \mathbb{R}^{d\times r}$ are respectively the down projection and up projection layers, with $r \ll \min(d, v)$. This adapter takes each value token at the l-layer as input and its output is added to the output of the pre-trained value projection layer (see notation $\oplus$ in Figure 3, lower left). Similarly the LoRA adapter for projection of key tokens takes each key token as input and its output is added to the output of the key projection layer.

C. Language Guidance on Visual Learning

We propose utilizing the powerful textual representation ability of the pre-trained VLM text encoder to guide the training of the visual adapters. When the image classifier learns c_t new visual classes at the t-th task, rich prior knowledge of each new class in the form of textual description is first obtained from a LLM model (e.g., ChatGPT [42]), and then encoded into the semantic textual space (i.e., output space of the VLM's text encoder) by the frozen text encoder. The distinctive semantic textual representation of each class in the unchanged textual space is used as the anchor to attract visual representations of the same class from the image encoder during training of the newly added visual adapters in the frozen image encoder. The detailed process is described below.

For each new visual class, multiple prompts are employed respectively as inputs to the LLM model to obtain multiple descriptive knowledge about the class. The prompt design is from the CuPL [46], including "Describe a [CLASS]", "Describe what a [CLASS] looks like", and "What are the identifying characteristics of the [CLASS]?", where '[CLASS]' denotes the name of the visual class. Multiple descriptive outputs (or sentences) are generated by the LLM model for each prompt. With multiple prompts from different aspects, it is expected that (at least part of) the multiple descriptive outputs from the LLM model would contain distinctive information about the visual class. Suppose totally M textual descriptions are generated by the LLM model for each class based on the prompts. For the k-th class, $k \in \{1, 2, \ldots, c_t\}$, after encoding the M textual descriptions of the class by the frozen text encoder, we can obtain the set of M textual representations of the k-th class in the semantic textual space, denoted by $\mathcal{G}_k = \{\mathbf{g}_{k,1}, \mathbf{g}_{k,2}, \ldots, \mathbf{g}_{k,M}\}$. Considering that each textual representation may contain only part of class-relevant information from certain aspect, all the M textual representations are aggregated to obtain the overall representation $\bar{\mathbf{g}}_k$ (aka 'textual anchor') of the k-th class with certain aggregation function $\bar{\mathbf{g}}_k = \pi(\mathcal{G}_k)$. For simplicity, the average function is used for aggregation here. Because of the strong representation power of the text encoder, the class-distinctive information within multiple textual descriptions is expected to be largely preserved in the textual anchor $\bar{\mathbf{g}}_k$, such that the textual anchor of each class (including both old and new classes) is distinctive in the semantic textual space.

In addition, in order to help the outputs of the image encoder for each visual class k more compactly clustered around the textual anchor $\bar{\mathbf{g}}_k$ of the same class k, fake OOD anchors in the textual space can be generated based on the textual representation sets $\{\mathcal{G}_k\}$ of all visual classes. Although textual

representation sets of old classes from previous learning rounds can also be included for fake OOD anchor generation, to avoid the storage of old sets, only the representation sets of new classes at the current t-th round are used. Denote by $\mathcal{O}_t$ the set of fake OOD anchors generated by certain fake OOD set generation function $\psi(\{\mathcal{G}_k\})$. While the function $\psi(\{\mathcal{G}_k\})$ can be carefully designed to generate more effective fake OOD anchors for each class, we leave the function design as future work and adopt a naive way to generate the fake OOD set $\mathcal{O}_t$, where one fake OOD anchor $\mathbf{o}_m$ is generated simply by collecting one textual representation per class and then averaging the collection, i.e., $\mathbf{o}_m = \frac{1}{c_t}\sum_k^{c_t} \mathbf{g}_{k,m}$ (also see Figure 3, lower right). In this way, M fake OOD anchors are generated and will be used to help train the visual adapters. Even with such simple average strategy, generated fake OOD anchors can already help improve the model performance both in image classification and OOD detection (see Section IV-C later).

With the help of the textual anchors $\{\bar{\mathbf{g}}_1, \bar{\mathbf{g}}_2, \ldots, \bar{\mathbf{g}}_{c_t}\}$ and fake OOD anchors $\mathcal{O}_t = \{\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_M\}$, the LoRA adapters specifically for the c_t new visual classes can be optimized using the well-known contrastive learning loss. Formally, let $\mathcal{D}_t = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$ represent the training set for the c_t new visual classes, with $\mathbf{x}_n$ denoting the n-th training image and $y_n \in \{1, 2, \ldots, c_t\}$ the corresponding class label. Denote by f_t the adapted image encoder (i.e., the original frozen VLM image encoder plus the inserted LoRA adapters), and $f_t(\mathbf{x}_n)$ the output of the image encoder for the input image $\mathbf{x}_n$. Then the contrastive learning loss is

$$\mathcal{L} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{c_t} \mathbb{1}(y_n = k)\left[\log \frac{\exp\{s(f_t(\mathbf{x}_n), \bar{\mathbf{g}}_k)/\tau\}}{Z(\mathbf{x}_n)}\right], \tag{2}$$

where $\mathbb{1}(\cdot)$ is the indicator function, $s(\cdot,\cdot)$ denotes the cosine similarity measurement, and τ is a temperature hyper-parameter. $Z(\mathbf{x}_n) = \sum_{j=1}^{c_t}\exp\{s(f_t(\mathbf{x}_n), \bar{\mathbf{g}}_j)/\tau\} + \sum_{m=1}^{M}\exp\{s(f_t(\mathbf{x}_n), \mathbf{o}_m)/\tau\}$. By minimizing the loss $\mathcal{L}$, each image $\mathbf{x}_n$ is attracted to the corresponding anchor $\bar{\mathbf{g}}_k$ of its class (i.e., $k = y_n$), while pushed away from the textual anchors of all the other classes and all the fake OOD anchors. Once the LoRA adapters are well-trained, all the training images of class k can be fed to the adapted image encoder f_t and the corresponding outputs are averaged to obtain the visual prototype $\mathbf{p}_{t,k}$ for class k. The visual prototypes of all (both new and old) classes will be used for model inference.

D. Model Inference

After learning the c_t new classes at the t-th round of continual learning, the model can be used to predict the class of any test image $\mathbf{z}$ if the image comes from one of the $C_t = c_1 + c_2 + \ldots + c_t$ learned classes, or to detect whether the test image is OOD (i.e., not from any of the learned classes). Both the visual prototypes and textual anchors of all learned classes are used for model inference. Specifically, the degree of the test image $\mathbf{z}$ belonging to the k-th class of the t-th task can be measured by

$$\mu_{t,k} = s_1(f_t(\mathbf{z}), \bar{\mathbf{g}}_k) + \lambda \cdot s_2(f_t(\mathbf{z}), \mathbf{p}_{t,k}), \tag{3}$$

where $s_1(\cdot,\cdot)$ and $s_2(\cdot,\cdot)$ are two similarity measurement functions and, for simplicity, cosine similarity is adopted for both functions. λ is a constant coefficient to balance the contribution of two similarities. Similarly, the degree of $\mathbf{z}$ belonging to any class of one previous (e.g., 1st, 2nd, $(t-1)$-th) task can be measured as by Equation 3, except that the LoRA adapters for the corresponding previous task are used during image feature extraction. The highest degree μ^* over all learned classes can be obtained by

$$\mu^* = \max_t \max_{k\in\{1,2,\ldots,c_t\}} \mu_{t,k}, \tag{4}$$

The class associated with μ^* is the final prediction for $\mathbf{z}$.

E. Comparison With Previous Works

Moreover, the proposed framework has several advantages over existing pre-trained VLM-based methods (i.e., AttriCLIP [34], PROOF [35], CIL-PVLM [33]). First, unlike AttriCLIP, PROOF, and CIL-PVLM, which use a single fixed template ("A photo of [CLASS NAME]"), the proposed framework designs multiple prompts to query the large language model, obtaining richer class information that assists classification. Second, different from PROOF and CIL-PVLM, this framework does not require replaying old class samples. Third, compared to AttriCLIP, PROOF, and CIL-PVLM, which only add learnable parameters at the feature vector level to adapt to downstream tasks, this framework uses parameter-efficient fine-tuning during feature extraction, better adapting to downstream tasks, especially when there is a significant domain gap between pre-training data and downstream task data. Additionally, each task in this framework has its own task-specific visual adapters, completely avoiding the plasticity-stability dilemma. Furthermore, unlike these three methods, this framework uses fake OOD anchors to improve continual learning performance and OOD detection performance. Finally, the framework effectively integrates textual and visual information for inference in a simple manner, further improving continual learning performance.

IV. Experiments

A. Experimental Setup

1) Datasets: Four datasets were used to evaluate the proposed framework, including CIFAR100 [47], ImageNet-R [48], Mini-ImageNet100 [49], and Skin40 [50], [51]. CIFAR100 consists of 100 categories of natural images, each category containing 500 images for training and 100 for testing. ImageNet-R presents a diverse set of 200 categories that include various styles such as cartoons, graffiti, and more challenging examples. It encompasses 30,000 images, with 24,000 designated for training and 6,000 for testing, and exhibits an imbalanced distribution of samples across categories. Mini-ImageNet100, crafted from ImageNet1k [49], is a natural image dataset featuring 100 classes, providing 1,000 training images and 200 testing images per class. Skin40

TABLE I
STATISTICS OF FOUR DATASETS. 'BALANCE' INDICATES WHETHER THE DATA IS BALANCED FOR EACH CLASS OF THE DATASET. '[19, 7016]', '[24, 6000]', AND '[420, 1640]': THE RANGE OF IMAGE WIDTH AND HEIGHT OF IMAGENET-R, MINI-IMAGENET100, AND SKIN40, RESPECTIVELY

Dataset	Classes	Train set	Test set	Balance	Number of tasks	Size
CIFAR100 [47]	100	50,000	10,000	✓	5, 10, 20	32×32
ImageNet-R [48]	200	24,000	6,000	×	5, 10, 20	[19, 7016]
Mini-ImageNet100 [49]	100	100,000	20,000	✓	5, 10	[24, 6000]
Skin40 [50], [51]	40	2,000	400	✓	5, 10	[420, 1640]

is a subset of 198 skin disease classes collected from the Internet and consists of 40 classes, with each class containing 50 images for training and 10 images for testing. Each dataset was partitioned into multiple subsets, each of which contains the same number of unique classes and is used as one learning task in the whole process of continual learning on the dataset. Specifically, CIFAR100 was segmented into sets of 5, 10, or 20 tasks; ImageNet-R was similarly divided into 5, 10, or 20 task groupings, while Mini-ImageNet100 and Skin40 were organized into 5 or 10 task configurations, respectively. The detailed statistical information for the four datasets is shown in Table I.

2) Implementation Details: For all experiments, CLIP's ViT-B-16 pre-trained from OpenAI serves as the backbone. In the training phase, we employ the AdamW optimizer, setting the learning rate at 0.005 with adjustments via cosine annealing decay, a weight decay of 0.1, and a batch size of 64. GPT-3 [52] is queried using three prompts to create 30 text descriptions per class, i.e., $M = 30$. The LoRA's rank r is set to 24, and λ is set to 1. Each image was resized to 224×224 pixels.

3) Evaluation Metrics: For the balanced datasets CIFAR100 and Mini-ImageNet100, continual learning performance is evaluated using two classification accuracies on test dataset, namely 'Last-ACC' and 'Avg-ACC'. 'Last-ACC' represents the accuracy achieved by the model on all learned classes finishing the final task. 'Avg-ACC' is the average of model performance over all rounds (tasks), with the model performance at the t-th round measured by the accuracy on learned ($C_t = c_1 + \ldots + c_t$) classes so far. For the class-imbalanced dataset ImageNet-R, the mean class recall (MCR) instead of accuracy is used, resulting in the two metrics 'Last-MCR' and 'Avg-MCR'. Note that 'Last-MCR' and 'Avg-MCR' are respectively equivalent of 'Last-ACC' and 'Avg-ACC' for class-balanced datasets.

For OOD detection evaluation on each test dataset, at the t-th round, all the learned C_t classes so far are used as in-distribution (ID) classes, and the remaining classes to be learned at subsequent rounds are considered as OOD classes. With the degrees $\{\mu_{t,k}\}$ for all learned classes as logits (i.e., input to a softmax operator), the maximum softmax probability (MSP) method [53] is adopted for OOD detection, and the area under the receiver operating characteristic curve (AUC) can be obtained after finishing each round of continual learning, resulting in the 'Last-AUC' and 'Avg-AUC' as for 'Last-ACC' and 'Avg-ACC'. For each experiment, three runs with distinct random seeds were performed, and the average and standard deviation of the three results were reported with each metric.

4) Baselines: Existing methods requiring retaining old data (including iCaRL [15], DarkER++ [30], DynaER [18], WA [25], PROOF [35], and CIL-PVLM [33]) and those without needing old data (including L2P [36], DualPrompt [20], CODA-Prompt [21], and AttriCLIP [34]) are adopted for comparison. Additionally, the simple continual CLIP [54] that relies solely on the excellent generalization capability of CLIP itself was also used as a baseline. For methods that retain old data, the memory buffer size is set to 2000 for CIFAR100 and Mini-ImageNet100, 4000 for ImageNet-R, and 80 for Skin40. The same MSP method is used for OOD detection with each baseline method.

B. Efficacy Evaluation of the Method

1) Continual Learning: Figure 4, Table II, and Table III show the continual learning performance of the different methods on the CIFAR100, ImageNet-R, and Mini-ImageNet100 datasets under different settings, respectively. The results demonstrate that the continual learning performance of the proposed framework ('Ours') is significantly better than all baselines, even though some of them use retained old data during continual learning. For example, among the methods without retaining old data, our method outperforms the best baseline DualPrompt by 7.37% and 8.01% in Last-ACC on CIFAR100 and in Last-MCR on ImageNet-R for 10 tasks, respectively. In addition, it can be observed that the proposed framework achieves better results regardless of whether the dataset is balanced or not, and actually the continual learning performance improvement on imbalanced dataset ImageNet-R is even more obvious.

2) OOD Detection: To further demonstrate the improvement in OOD detection performance of the proposed framework compared to previous work, we evaluated the OOD detection performance of our method and previous methods on various settings of the CIFAR100, ImageNet-R, and Mini-ImageNet100 datasets. According to the results in Table IV and Table V, our method outperforms all baselines. For example, in various settings of ImageNet-R, our method is around 5-8% higher in Last-AUC than that of the best continual learning method (i.e., DualPrompt) without retaining old data. On the Mini-ImageNet100 dataset, our method exhibits outstanding performance with Last-AUCs of 96.29% and 94.56% for T=5 and T=10 settings, respectively, significantly outperforming other methods. All the results support that the proposed framework exhibits better OOD detection performance.

From these results on CIFAR100, ImageNet-R, and Mini-ImageNet100, it is clear that our method significantly outperforms all baselines in continual learning and additionally in OOD detection.

C. Ablation Study

Extensive ablation studies were performed to confirm the effect of each component or operation in the proposed framework. According to the results shown in Table VI, the

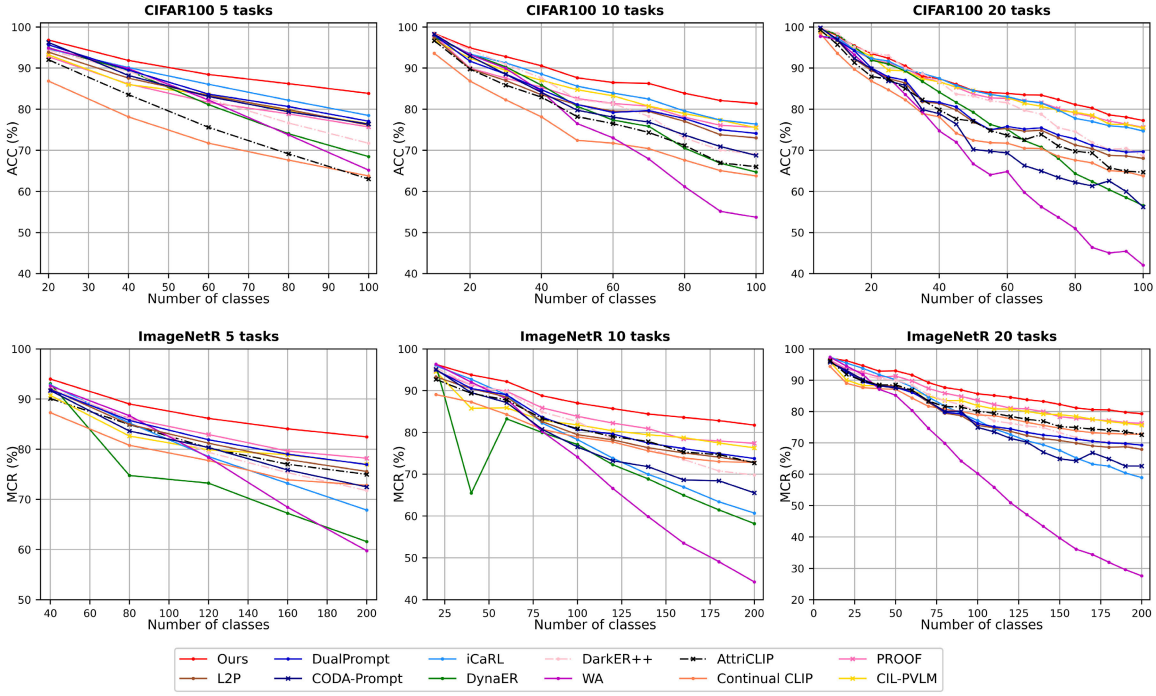


Fig. 4. Performance of continual learning on the CIFAR100 and ImageNet-R dataset, respectively. Standard deviation (with range (0.03, 2.89]) is ignored for clear presentation.

TABLE II

CONTINUAL LEARNING PERFORMANCE OF DIFFERENT METHODS ON CIFAR100 AND IMAGENET-R WITH DIFFERENT NUMBER OF TASKS (I.E., T=5, 10, 20), RESPECTIVELY. 'MEMORY' INDICATES WHETHER DATA OF LEARNED CLASSES ARE REQUIRED

Method	Memory	CIFAR100						ImageNet-R					
		T = 5		T = 10		T = 20		T = 5		T = 10		T = 20	
		Last-ACC	Avg-ACC	Last-ACC	Avg-ACC	Last-ACC	Avg-ACC	Last-MCR	Avg-MCR	Last-MCR	Avg-MCR	Last-MCR	Avg-MCR
iCaRL [15]	✓	78.47±1.05	86.30±0.89	76.37±1.06	85.17±0.31	74.68±0.72	85.04±0.43	67.85±0.72	79.52±0.30	60.70±0.35	77.30±0.28	58.92±0.64	77.23±0.2 8
DynaER [18]	✓	68.47±0.46	81.59±0.27	64.72±0.90	80.31±0.33	56.59±1.73	78.10±0.30	61.57±1.04	73.96±0.56	58.18±0.69	72.76±0.65	46.29±0.83	67.58±0.12
DarkER++ [30]	✓	71.70±2.07	82.90±1.47	69.02±0.55	82.53±0.05	68.63±1.01	83.29±0.18	71.73±1.17	80.92±0.38	69.80±0.76	81.26±0.24	67.88±0.53	80.35±0.32
WA [25]	✓	65.18±0.75	81.18±0.78	53.73±1.86	75.30±1.00	42.06±0.77	68.50±0.31	59.76±0.34	77.17±0.26	44.24±0.16	70.41±0.32	27.58±0.41	60.09±0.26
PROOF [35]	✓	75.67±0.28	83.02±0.08	75.56±0.20	83.53±0.06	75.49±0.26	85.22±0.14	78.19±0.16	83.73±0.19	77.39±0.09	82.09±0.68	76.27±0.36	84.19±0.09
CIL-PVLM [33]	✓	76.44±0.35	83.77±0.24	75.47±0.89	84.61±0.56	75.41±0.81	85.01±0.35	76.97±0.08	81.81±0.86	76.31±0.08	82.34±0.64	75.67±0.12	82.77±0.61
L2P [36]	×	76.13±0.03	84.16±0.17	73.01±0.12	82.17±0.11	68.03±0.30	79.59±0.17	75.55±0.11	82.32±0.09	72.86±0.28	81.10±0.25	67.94±0.19	78.28±0.21
DualPrompt [20]	×	77.07±0.44	85.31±0.17	74.12±0.15	82.98±0.08	69.66±0.73	80.28±0.32	76.94±0.31	83.08±0.02	73.76±0.14	82.08±0.03	69.30±0.30	78.85±0.49
CODA-Prompt [21]	×	76.37±0.59	84.65±0.06	68.76±0.21	81.18±0.36	56.21±2.41	74.65±0.55	72.47±0.33	80.79±0.09	65.54±0.66	77.67±0.27	62.62±0.83	76.43±0.25
AttriCLIP [34]	×	68.35±0.28	79.39±0.06	66±1.41	78.84±0.09	64.70±0.89	78.19±0.30	74.97±0.11	81.44±0.14	72.71±0.49	81.36±0.16	72.62±1.17	81.28±0.25
Continual CLIP [54]	×	63.77	73.62	63.77	76.42	63.77	76.02	72.80	78.50	72.80	79.34	72.80	80.02
Ours	×	**83.84±0.17**	**89.44±0.11**	**81.49±0.15**	**88.44±0.20**	**77.26±0.59**	**86.40±0.07**	**82.46±0.31**	**87.14±0.42**	**81.77±0.12**	**87.65±0.29**	**79.32±0.25**	**86.77±0.23**

TABLE III

CONTINUAL LEARNING PERFORMANCE OF DIFFERENT METHODS ON MINI-IMAGENET100 WITH THE DIFFERENT NUMBER OF TASKS (I.E., T=5, 10). 'MEMORY' INDICATES WHETHER DATA OF LEARNED CLASSES ARE REQUIRED

Method	Memory	T = 5		T = 10	
		Last-ACC	Avg-ACC	Last-ACC	Avg-ACC
iCaRL [15]	✓	91.36±0.35	94.90±0.24	89.70±0.18	94.59±0.15
DynaER [18]	✓	88.67±0.50	93.84±0.28	85.82±0.21	92.82±0.08
DarkER++ [30]	✓	81.35±0.52	89.74±0.26	75.68±1.19	87.84±0.33
WA [25]	✓	82.66±0.79	91.41±0.24	73.93±1.45	88.76±0.73
PROOF [35]	✓	92.84±0.06	95.48±0.03	92.81±0.05	95.83±0.04
CIL-PVLM [33]	✓	93.68±0.11	96.03±0.14	92.71±0.18	94.50±0.09
L2P [36]	×	92.36±0.13	95.21±0.06	90.97±0.28	94.94±0.13
DualPrompt [20]	×	92.49±0.17	95.25±0.11	91.46±0.21	95.21±0.09
CODA-Prompt [21]	×	92.56±0.16	95.40±0.06	89.29±0.53	94.34±0.15
AttriCLIP [34]	×	87.67±1.15	93.05±0.41	86.24±0.94	91.74±0.46
Continual CLIP [54]	×	89.99	92.65	89.99	93.13
Ours	×	**94.07±0.12**	**96.13±0.03**	**93.91±0.18**	**96.53±0.09**

performance of continual learning is gradually improved as more components are added, which proves the effectiveness of each proposed component. Specifically, in Table VI, the results from the 2nd and 3rd columns demonstrate the effectiveness of task-specific visual adapters which help solve the plasticity-stability dilemma. The results in the 3rd and 4th columns indicate that even a single fake OOD anchor can improve continual learning performance, such as a 0.7% improvement on Skin40, fake OOD anchor help each visual class cluster more compactly around its corresponding text anchor. At the same time, the results in the 5th and 6th columns show that multiple fake OOD anchors constructed can further enhance continual learning performance, such as a 0.97% improvement on CIFAR100, further proving the effectiveness of fake OOD anchors. The results in the 3rd and 5th columns present that multiple text descriptions can improve performance, such as a 1.00% improvement on Skin40, proving that multiple text descriptions can provide more class information and help classification. Moreover, the results in the 4th and 6th

TABLE IV

OOD DETECTION PERFORMANCE OF DIFFERENT METHODS ON CIFAR100 AND IMAGENET-R WITH DIFFERENT NUMBER OF TASKS (I.E., T=5, 10, 20), RESPECTIVELY. 'MEMORY' INDICATES WHETHER DATA OF LEARNED CLASSES ARE REQUIRED

Method	Memory	CIFAR100						ImageNet-R					
		T = 5		T = 10		T = 20		T = 5		T = 10		T = 20	
		Last-AUC	Avg-AUC	Last-AUC	Avg-AUC	Last-AUC	Avg-AUC	Last-AUC	Avg-AUC	Last-AUC	Avg-AUC	Last-AUC	Avg-AUC
iCaRL [15]	✓	79.92±0.63	82.51±0.45	78.73±0.49	82.32±0.42	76.73±0.54	81.30±0.32	73.91±0.10	80.85±0.28	69.06±0.69	79.25±0.22	66.43±1.50	78.15±0.37
DynaER [18]	✓	59.39±0.43	70.18±0.48	61.53±2.37	68.15±0.59	56.78±0.53	64.00±0.27	66.35±1.25	72.20±0.30	63.16±2.47	71.56±0.56	56.21±1.62	66.89±0.22
DarkER++ [30]	✓	73.65±1.12	79.32±1.33	74.24±1.49	77.89±0.29	69.77±0.66	78.31±0.63	77.94±0.69	81.42±0.42	72.16±1.88	80.88±0.45	68.98±3.35	79.83±0.15
WA [25]	✓	69.59±0.86	76.82±0.37	64.37±1.73	72.36±0.38	51.50±2.88	68.30±0.42	73.98±0.72	81.08±0.21	64.39±1.28	76.90±0.15	55.37±3.05	71.08±0.26
PROOF [35]	✓	71.28±0.25	75.81±0.10	73.09±0.37	76.18±0.18	71.21±0.50	76.04±0.11	77.82±0.20	79.37±0.18	75.99±0.14	79.14±0.03	72.61±0.29	78.87±0.09
CIL-PVLM [33]	✓	72.80±0.25	77.79±0.35	71.64±0.28	78.02±0.71	69.09±0.41	78.14±0.91	77.16±0.01	79.90±1.08	75.88±1.12	79.95±1.28	75.93±1.09	79.97±0.98
L2P [36]	×	75.52±0.17	79.56±0.07	78.91±0.52	81.06±0.25	81.99±0.19	82.38±0.34	79.59±0.13	82.22±0.05	76.05±0.44	81.82±0.17	72.00±0.92	80.37±0.22
DualPrompt [20]	×	76.60±0.46	80.92±0.12	79.56±0.55	81.86±0.26	82.11±0.56	82.98±0.25	80.95±0.14	83.48±0.06	76.12±0.10	82.49±0.08	72.65±0.73	81.08±0.20
CODA-Prompt [21]	×	77.16±0.89	82.53±0.06	76.36±1.72	80.97±0.11	77.81±0.95	80.29±0.36	78.75±0.04	82.88±0.14	75.53±0.67	81.04±0.39	67.62±1.68	79.80±0.17
AttriCLIP [34]	×	72.39±1.14	77.62±0.24	70.85±3.47	77.16±0.68	71.64±0.66	76.75±0.13	79.44±0.83	81.44±0.14	77.29±0.29	81.60±0.28	72.30±1.73	80.79±0.31
Continual CLIP [54]	×	72.28	77.74	72.28	78.43	72.28	78.72	78.10	80.96	78.10	80.97	78.10	80.80
Ours	×	**85.59±0.25**	**89.77±0.06**	**87.11±0.28**	**89.41±0.38**	**82.72±1.23**	**87.55±0.42**	**86.12±0.11**	**88.96±0.09**	**83.97±0.36**	**89.17±0.14**	**78.71±0.28**	**88.20±0.07**

TABLE V

OOD DETECTION PERFORMANCE OF DIFFERENT METHODS ON MINI-IMAGENET100 WITH THE DIFFERENT NUMBER OF TASKS (I.E., T=5, 10). 'MEMORY' INDICATES WHETHER DATA OF LEARNED CLASSES ARE REQUIRED

Method	Memory	$T = 5$		$T = 10$	
		Last-AUC	Avg-AUC	Last-AUC	Avg-AUC
iCaRL [15]	✓	88.01±0.37	91.95±0.12	80.00±0.92	88.88±0.17
DynaER [18]	✓	74.54±0.47	81.63±0.49	69.02±1.23	78.95±0.39
DarkER++ [30]	✓	80.10±1.07	88.14±0.41	72.27±0.94	83.80±0.42
WA [25]	✓	78.34±1.05	86.68±0.27	70.02±2.53	81.91±1.45
PROOF [35]	✓	85.34±0.26	87.52±0.02	85.74±0.82	87.64±0.23
CIL-PVLM [33]	✓	85.20±0.21	89.41±0.01	82.29±0.21	85.00±0.15
L2P [36]	×	88.87±0.30	91.32±0.11	85.02±0.49	90.89±0.11
DualPrompt [20]	×	89.74±0.29	91.88±0.08	85.77±0.58	91.47±0.18
CODA-Prompt [21]	×	90.05±0.34	92.02±0.14	82.58±0.61	90.64±0.18
AttriCLIP [34]	×	84.65±1.21	89.36±0.16	81.87±1.54	87.92±0.36
Continual CLIP [54]	×	93.66	95.16	93.66	95.13
Ours	×	**96.29±0.13**	**96.76±0.07**	**94.56±0.32**	**96.77±0.17**

TABLE VI

ABLATION STUDY OF THE PROPOSED FRAMEWORK ON CIFAR100 (10 TASKS), IMAGENET-R (10 TASKS), AND SKIN40 (10 TASKS). THE RANGE OF STANDARD DEVIATION IS [0.15, 1.23]

Task-shared visual adapters		✓				
Task-specific visual adapters			✓	✓	✓	✓
Fake OOD anchor				✓		✓
Multiple text descriptions					✓	✓
Last-ACC (CIFAR100)	63.77	24.43	80.40	80.96	80.52	**81.49**
Last-MCR (ImageNet-R)	72.80	26.71	81.42	81.65	81.68	**81.77**
Last-ACC (Skin40)	14.75	12.33	49.50	50.25	50.50	**51.42**

TABLE VII

ABLATION STUDY ON SIMILARITY MEASURES DURING MODEL INFERENCE. T=10 FOR BOTH CIFAR100 AND IMAGENET-R

Inference		CIFAR100		ImageNet-R	
s_1	s_2	Last-ACC	Avg-ACC	Last-MCR	Avg-MCR
✓		77.24±0.58	85.93±0.08	80.33±0.35	86.50±0.22
	✓	80.10±0.51	87.62±0.36	78.26±0.34	85.74±0.17
✓	✓	**81.49±0.15**	**88.44±0.20**	**81.77±0.12**	**87.65±0.29**

columns further demonstrate the effectiveness of multiple text descriptions, such as a 1.17% improvement on Skin40.

According to the results in Table VII, the proposed framework can already obtain good continual learning performance when using either of the two similarity measures (s_1 and s_2) during inference. This may be attributed to the proposed language guidance of the visual adapter training through the textual knowledge in the unchanged textual space. Combining the two similarity measures results in the best performance (last row), confirming that integrating information from both image and text modalities can further improve model learning during inference. In addition, according to Figure 5, it can be observed that on both datasets, usage of fake OOD anchors clearly improves the model performance in both continual learning and OOD detection.

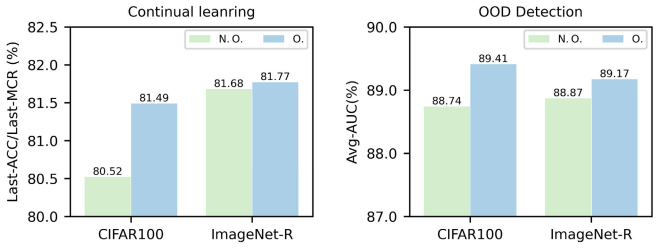


Fig. 5. Ablation study of fake OOD anchors for both continual learning (left) and OOD detection (right). Ten tasks are continually learned on CIFAR100 and ImageNet-R, respectively. 'O.' and 'N.O.' represent using and not using fake OOD anchors, respectively.

D. Sensitive and Generalizability Studies

1) Sensitive Study: To evaluate the influence of the hyper-parameter r (see Equation 1 and relevant description) in task-specific LoRA adapters on the performance of the proposed framework, a sensitivity study was performed by setting r respectively to {8, 10, 16, 20, 24, 30, 32}. According to the results shown in Figure 6 (left), Last-ACC is highest at 81.49 when r is set to 24 and lowest at 80.74 when r is set to 10. Changes in the hyper-parameter r of task-specific LoRA adapters (see Equation 1 and relevant description) only bring less than 1% fluctuation in Last-ACC. Therefore, the performance of the proposed framework is largely robust to the value choice of the hyper-parameter r.

Furthermore, since the two similarity measures during inference are important in our method, we perform a sensitivity study to show the effect of λ in Equation 3. λ is set respectively to {0.5, 0.7, 0.9, 1, 1.1, 1.3, 1.5}. The results shown in Figure 6 (right) confirm the proposed framework performs very stable for changes in the hyper-parameter λ within a certain range.

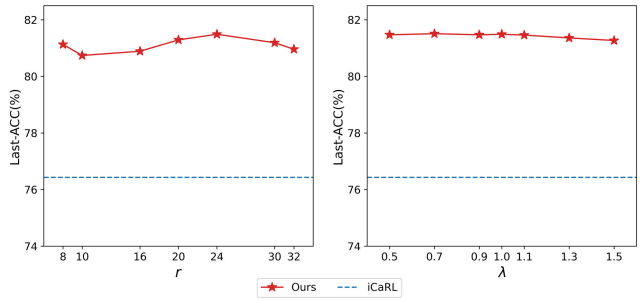


Fig. 6. Sensitivity study of rank r of task-specific LoRA adapters and λ in Equation 3 on CIFAR100 (10 tasks).

TABLE VIII

FEW-SHOT CIL PERFORMANCE (LAST-ACC) ON CIFAR100, WITH 10 ROUNDS OF CONTINUAL LEARNING PERFORMED AT EACH FEW-SHOT SETTING

Method	1 shot	2 shot	4 shot	8 shot	16 shot
L2P [36]	27.16±1.42	40.47±1.22	50.47±1.12	56.44±1.05	61.44±0.52
DualPrompt [20]	28.11±2.11	41.98±1.67	51.43±0.42	59.18±0.63	62.82±1.12
CODA-Prompt [21]	32.67±0.87	43.83±0.18	53.01±0.35	59.87±1.17	63.85±1.21
Linear-CLIP [23]	28.83±0.58	38.15±1.37	45.63±0.94	49.47±0.90	54.13±0.95
Continual CLIP [54]	63.77	63.77	63.77	63.77	63.77
AttriCLIP [34]	62.36±0.96	63.55±0.29	63.78±0.52	64.05±0.38	64.24±0.09
Ours	**67.13**±0.07	**68.57**±0.09	**69.11**±0.04	**72.61**±0.15	**74.31**±0.09

TABLE IX

CONTINUAL LEARNING PERFORMANCE OF DIFFERENT METHODS ON SKIN40 WITH THE DIFFERENT NUMBER OF TASKS (I.E., T=5, 10). 'MEMORY' INDICATES WHETHER DATA OF LEARNED CLASSES ARE REQUIRED

Method	Memory	Skin40			
		T = 5		T = 10	
		Last-ACC	Avg-ACC	Last-ACC	Avg-ACC
iCaRL [15]	✓	39.92±1.88	67.29±0.87	36.13±2.30	66.11±2.10
DynaER [18]	✓	31.42±1.59	65.78±1.44	23.33±1.76	59.45±0.47
DarkER++ [30]	✓	32.00±0.87	61.64±0.09	25.42±1.66	57.58±1.09
WA [25]	✓	27.88±0.53	59.39±0.04	20.63±0.53	52.12±0.01
PROOF [35]	✓	53.00±0.35	73.00±0.62	50.08±0.13	71.07±0.18
CIL-PVLM [33]	✓	41.25±0.86	52.74±0.59	39.25±0.35	47.14±0.55
L2P [36]	×	43.92±0.63	66.57±0.97	37.25±1.25	58.69±0.78
DualPrompt [20]	×	42.17±1.28	66.97±0.75	37.5±1.32	61.26±0.88
CODA-Prompt [21]	×	40.17±0.88	62.37±0.34	30.25±2.38	54.84±1.06
AttriCLIP [34]	×	19.92±1.06	47.62±1.59	13.92±2.27	39.86±1.89
Continual CLIP [54]	×	14.75	28.85	14.75	31.93
Ours	×	**54.67**±0.29	**72.89**±0.15	**51.42**±1.23	**69.12**±0.89

2) Generalizability Study: In addition, to further evaluate the generalization of the proposed framework, the proposed framework was performed on different settings of the skin40 dataset (i.e., T = 5, 10). In this case, textual descriptions generated by ordinary LLM may be unreliable. Therefore the textual descriptions of the skin40 dataset are generated by consulting the Internet. According to the results shown in Table IX, the continual learning performance of the proposed framework is largely superior to all baselines, proving its strong generalization ability even if the current data is significantly different from the pre-training data domain.

Under the 10 tasks setting on CIFAR100, few-shot class-incremental learning was performed to evaluate the generalization ability of the proposed framework. Specifically, at each round of continual learning, the number of training images per class was respectively set to 1, 2, 4, 8, and 16. As Table VIII shows, the proposed framework performs clearly better than all baselines, proving that it can be effectively generalized to few-shot CIL scenarios.

TABLE X

CONTINUAL LEARNING PERFORMANCE (LAST-ACC/MCR) OF DIFFERENT METHODS BASED ON CLIP AND OPENCLIP UNDER THE 10-TASK SETTINGS ON IMAGENET-R AND SKIN40. NOTE THAT CIL-PVLM AND PROOF REQUIRE STORING SMALL DATA FOR EACH OLD CLASS. 'ZS' DENOTES PERFORMANCE OF ZERO-SHOT OF PRE-TRAINED VLM. THE RANGE OF STANDARD DEVIATION IS [0.15, 1.24]

Dataset		CLIP				OpenCLIP			
		ZS	PROOF	CIL-PVLM	Ours	ZS	PROOF	CIL-PVLM	Ours
ImageNet-R	Last-MCR	72.80	77.39	75.24	**81.77**	81.56	82.45	81.39	**83.77**
	Avg-MCR	79.34	84.26	82.09	**87.65**	86.34	88.91	86.84	**88.92**
Skin40	Last-ACC	14.75	50.08	39.25	**51.42**	25.5	56.42	45.38	**59.15**
	Avg-ACC	31.93	71.07	47.14	**69.12**	41.83	73.34	49.27	**73.54**

TABLE XI

CONTINUAL LEARNING PERFORMANCE (LAST-ACC) OF DIFFERENT METHODS BASED ON MEDCLIP UNDER THE 10-TASK SETTINGS ON SKIN40. NOTE THAT CIL-PVLM AND PROOF REQUIRE STORING SMALL DATA FOR EACH OLD CLASS. 'ZS' DENOTES PERFORMANCE OF ZERO-SHOT OF PRE-TRAINED VLM. THE RANGE OF STANDARD DEVIATION IS [0.15, 0.45]

Dataset		MedCLIP			
		ZS	PROOF	CIL-PVLM	Ours
Skin40	Last-ACC	1.50	22.67	2.63	**43.88**
	Avg-ACC	6.39	49.96	14.85	**64.57**

Moreover, we conducted multiple experiments on different pre-trained VLMs to evaluate whether the proposed framework can generalize well across various pre-trained VLMs. Table X presents a comparison of continual learning performance between our method, CIL-PVLM, and PROOF on CLIP [23] and OpenCLIP [55] under the 10-task settings on ImageNet-R and Skin40. CLIP and OpenCLIP are VLMs pre-trained on no less than 4 million natural data. According to the results shown in Table X, all three methods show improved continual learning performance on OpenCLIP, with our method performing the best. This demonstrates that VLM-based continual learning methods can achieve better performance with stronger pre-trained VLMs, proving the superiority of the proposed framework. In addition, Table XI presents a comparison of continual learning performance between our method, CIL-PVLM, and PROOF on MedCLIP [56] under the 10-task settings on Skin40. MedCLIP [56] is a VLM pre-trained on medical data (i.e., less than 600K X-ray data). According to the results shown in Table XI, our method's continual learning performance in Last-ACC surpasses MedCLIP's Zero-Shot, PROOF, and CIL-PVLM by 42.38%, 21.21%, and 41.25%, respectively. This indicates that our method maintains good performance even when the pre-training data is limited and there is a significant domain gap between the pre-training data and the downstream task data, further confirming the superiority of the proposed framework.

E. Inference Time Comparison

According to the results shown in Table XII, it can be observed that the inference time for a single test image using the proposed framework is comparable to existing methods in the initial stages of continual learning. However, as the number

TABLE XII
INFERENCE TIME OF DIFFERENT METHODS FOR A SINGLE TEST IMAGE IN CIFAR100 WITH 10 TASKS. 'MEMORY' INDICATES WHETHER DATA OF LEARNED CLASSES ARE REQUIRED

Method	Memory	Inference time for a single test image (millisecond)									
		Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8	Task 9	Task 10
iCaRL [15]	✓	4.21	4.47	4.34	4.18	4.27	4.41	4.10	4.12	4.10	4.11
DynaER [18]	✓	1.80	2.84	3.80	4.87	6.01	7.05	8.21	9.21	10.35	11.44
DarkER++ [30]	✓	2.22	2.96	2.84	2.68	2.79	2.68	2.69	2.76	2.74	2.69
WA [25]	✓	1.54	1.55	1.43	1.48	1.37	1.35	1.35	1.36	1.36	1.39
PROOF [35]	✓	1.54	1.55	1.43	1.48	1.37	1.35	1.35	1.36	1.36	1.39
CIL-PVLM [33]	✓	4.91	4.18	4.12	4.14	4.51	4.32	4.40	4.36	4.30	4.07
L2P [36]	×	4.96	5.13	4.83	4.45	4.96	4.96	4.98	5.00	4.92	4.94
DualPrompt [20]	×	2.97	2.70	2.65	2.63	2.60	2.60	2.62	2.61	2.59	2.60
CODA-Prompt [21]	×	2.95	2.70	2.63	2.60	2.58	2.57	2.58	2.57	2.56	2.55
AttriCLIP [34]	×	3.03	4.92	6.87	8.73	10.70	12.65	14.77	16.44	18.54	20.43
Ours	×	3.76	4.01	5.11	6.58	7.56	8.77	10.00	11.24	12.57	13.89

of continual learning tasks increases, the inference time does become longer, with the proposed framework performing only slightly better than AttriCLIP. This is because the framework employs task-specific visual adapters to address the plasticity-stability dilemma, effectively enhancing continual learning performance. However, during inference, it requires sequential forward passes through each adapted image encoder (i.e., task-specific visual adapters and the frozen pre-trained VLM's image encoder), leading to increased inference time. Nonetheless, this inference time is still acceptable as it increases linearly with the number of training rounds.

V. CONCLUSION

In this study, a novel continual learning framework based on a pre-trained vision-language model is proposed. This framework ingeniously utilizes the fixed textual space to guide the continual learning of visual classes. The rich semantic knowledge for each visual class can be obtained from a LLM and obtained by the frozen VLM's text encoder, which is then utilized to guide the training of task-specific visual adapters. The usage of fake OOD textual representations during training and integration of multi-modality features during inference further improve the performance particularly in continual learning. The proposed framework achieves superior performance on multiple datasets under various continual learning settings. In future work, the proposed framework will be extended to solve more types of tasks, including continual medical image classification, fine-grained classification, and continual segmentation and detection tasks.

REFERENCES

[1] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–12.

[2] S. Liu et al., "Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection," 2023, *arXiv:2303.05499*.

[3] S. Hao, Y. Zhou, Y. Guo, R. Hong, J. Cheng, and M. Wang, "Real-time semantic segmentation via spatial-detail guided context propagation," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–12, Aug. 2022.

[4] G. Xu, J. Li, G. Gao, H. Lu, J. Yang, and D. Yue, "Lightweight real-time semantic segmentation network with efficient transformer and CNN," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 15897–15906, Jul. 2023.

[5] S. Wang, Z. Wang, H. Li, J. Chang, W. Ouyang, and Q. Tian, "Semantic-guided information alignment network for fine-grained image recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 11, pp. 6558–6570, Sep. 2023.

[6] L. Ouyang et al., "Training language models to follow instructions with human feedback," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 1–15.

[7] A. J. Thirunavukarasu et al., "Large language models in medicine," *Nature Med.*, vol. 29, no. 8, pp. 1930–1940, 2023.

[8] P. Georgieva and P. Zhang, "Optical character recognition for autonomous stores," in *Proc. IEEE 10th Int. Conf. Intell. Syst. (IS)*, Aug. 2020, pp. 69–75.

[9] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," *Psychol. Learn. Motivat.*, vol. 24, pp. 109–165, 1989.

[10] J. Kirkpatrick et al., "Overcoming catastrophic forgetting in neural networks," *Proc. Nat. Acad. Sci. USA*, vol. 114, no. 13, pp. 3521–3526, Mar. 2017.

[11] T. Konishi, M. Kurokawa, C. Ono, Z. Ke, G. Kim, and B. Liu, "Parameter-level soft-masking for continual learning," in *Proc. Int. Conf. Mach. Learn.*, 2023, pp. 1–14.

[12] W. Cong, Y. Cong, G. Sun, Y. Liu, and J. Dong, "Self-paced weight consolidation for continual learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 4, pp. 2209–2222, Apr. 2024.

[13] H. Ahn, J. Kwak, S. Lim, H. Bang, H. Kim, and T. Moon, "SS-IL: Separated softmax for incremental learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Jun. 2021, pp. 1–10.

[14] X. Chen and X. Chang, "Dynamic residual classifier for class incremental learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 18743–18752.

[15] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "ICaRL: Incremental classifier and representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5533–5542.

[16] S. Cha, S. Cho, D. Hwang, S. Hong, M. Lee, and T. Moon, "Rebalancing batch normalization for exemplar-based class-incremental learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 20127–20136.

[17] S. Hou, X. Pan, C. C. Loy, Z. Wang, and D. Lin, "Learning a unified classifier incrementally via rebalancing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 831–839.

[18] S. Yan, J. Xie, and X. He, "DER: Dynamically expandable representation for class incremental learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1–10.

[19] X. Liu et al., "P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, 2022, pp. 61–68.

[20] Z. Wang et al., "DualPrompt: Complementary prompting for rehearsal-free continual learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 631–648.

[21] J. S. Smith et al., "CODA-Prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 1–11.

[22] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *Proc. Int. Conf. Mach. Learn.*, 2023, pp. 19730–19742.

[23] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 1–16.

[24] E. J. Hu et al., "LoRA: Low-rank adaptation of large language models," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–13.

[25] B. Zhao, X. Xiao, G. Gan, B. Zhang, and S.-T. Xia, "Maintaining discrimination and fairness in class incremental learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13205–13214.

[26] F. Benzing, "Unifying importance based regularisation methods for continual learning," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2022, pp. 1–25.

[27] Z. Li, C. Zhong, S. Liu, R. Wang, and W.-S. Zheng, "Preserving earlier knowledge in continual learning with the help of all previous feature extractors," 2021, *arXiv:2104.13614*.

[28] B. Ni et al., "MoBoo: Memory-boosted vision transformer for class-incremental learning," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–15, Jun. 2024.

[29] D.-W. Zhou, Q.-W. Wang, H.-J. Ye, and D.-C. Zhan, "A model or 603 exemplars: Towards memory-efficient class-incremental learning," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–35.

[30] P. Buzzega, M. Boschini, A. Porrello, D. Abati, and S. Calderara, "Dark experience for general continual learning: A strong, simple baseline," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 1–11.

[31] Z. Li, C. Zhong, R. Wang, and W.-S. Zheng, "Continual learning of new diseases with dual distillation and ensemble strategy," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.*, 2020, pp. 169–178.

[32] M. Kang, J. Park, and B. Han, "Class-incremental learning by knowledge distillation with adaptive feature consolidation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 1–10.

[33] X. Liu, X. Cao, H. Lu, J.-W. Xiao, A. D. Bagdanov, and M.-M. Cheng, "Class incremental learning with pre-trained vision-language models," 2023, *arXiv:2310.20348*.

[34] R. Wang et al., "AttriCLIP: A non-incremental learner for incremental knowledge learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 3654–3663.

[35] D.-W. Zhou, Y. Zhang, J. Ning, H.-J. Ye, D.-C. Zhan, and Z. Liu, "Learning without forgetting for vision-language models," 2023, *arXiv:2305.19270*.

[36] Z. Wang et al., "Learning to prompt for continual learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 139–149.

[37] T.-Y. Wu et al., "Class-incremental learning with strong pre-trained models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9591–9600.

[38] G. Zhang, L. Wang, G. Kang, L. Chen, and Y. Wei, "SLCA: Slow learner with classifier alignment for continual learning on a pre-trained model," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 19148–19158.

[39] J. Yu, J. Li, Z. Yu, and Q. Huang, "Multimodal transformer with multi-view visual representation for image captioning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 12, pp. 4467–4480, Dec. 2020.

[40] W. Zhang, C. Ma, Q. Wu, and X. Yang, "Language-guided navigation via cross-modal grounding and alternate adversarial learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 9, pp. 3469–3481, Sep. 2021.

[41] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 1–12.

[42] OpenAI. (2019). *Introducing Chatgpt*. [Online]. Available: https://openai.com/blog/chatgpt

[43] N. Ding et al., "Parameter-efficient fine-tuning of large-scale pre-trained language models," *Nature Mach. Intell.*, vol. 5, no. 3, pp. 220–235, Mar. 2023.

[44] M. Dehghani et al., "Scaling vision transformers to 22 billion parameters," in *Proc. Int. Conf. Mach. Learn.*, 2023, pp. 7480–7512.

[45] S. Luo et al., "LCM-LoRA: A universal stable-diffusion acceleration module," 2023, *arXiv:2311.05556*.

[46] S. Pratt, I. Covert, R. Liu, and A. Farhadi, "What does a platypus look like? Generating customized prompts for zero-shot image classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 15691–15701.

[47] A. Krizhevsky et al., "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep. TR-2009, 2009.

[48] D. Hendrycks et al., "The many faces of robustness: A critical analysis of out-of-distribution generalization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 1–10.

[49] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[50] Z. Xu, K. Chen, W.-S. Zheng, Z. Tan, X. Yang, and R. Wang, "Expert with outlier exposure for continual learning of new diseases," in *Proc. IEEE Int. Conf. Bioinf. Biomed. (BIBM)*, Dec. 2022, pp. 1768–1772.

[51] Y. Yang, Z. Cui, J. Xu, C. Zhong, W.-S. Zheng, and R. Wang, "Continual learning with Bayesian model based on a fixed pre-trained feature extractor," *Vis. Intell.*, vol. 1, no. 5, pp. 1–14, 2023.

[52] T. Brown et al., "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 1877–1901.

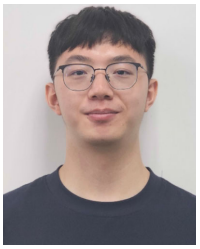
[53] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–12.

[54] V. Thengane, S. Khan, M. Hayat, and F. Khan, "CLIP model is an efficient continual learner," 2022, *arXiv:2210.03114*.

[55] M. Cherti et al., "Reproducible scaling laws for contrastive language-image learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 2818–2829.

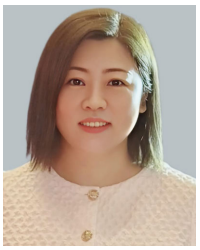
[56] Z. Wang, Z. Wu, D. Agarwal, and J. Sun, "MedCLIP: Contrastive learning from unpaired medical images and text," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2022, pp. 11–12.

Wentao Zhang received the B.S. and M.S. degrees in 2017 and 2020, respectively. He is currently pursuing Ph.D. degree with Sun Yat-sen University. His research interests include computer vision, medical image analysis, and multimodal learning.

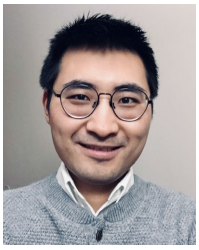


Yujun Huang received the B.S. degree from Guangdong University of Technology in June 2021. He is currently a pursuing the master's degree with Sun Yat-sen University. His research interests include machine learning and deep learning, especially on continual learning, pre-trained model reuse, and multimodal models.

Weizhuo Zhang received the B.S. degree in secrecy management from Sun Yat-sen University, China, in 2022, where he is currently pursuing the M.S. degree with the School of Computer Science and Engineering. His research interests include computer vision and machine learning.



Tong Zhang (Member, IEEE) received the B.E. and M.E. degrees from Harbin Institute of Technology, China, and the Ph.D. degree in information technology from The University of Sydney, Australia. She is currently an Assistant Professor with the Department of Network Intelligence, Peng Cheng Laboratory. Her research interests include multimodal foundation models, computer vision, and medical image analysis, with a particular focus on AI-aided clinical applications in prenatal and cardiac diagnosis and treatment.



Qicheng Lao received the B.S. degree in medicine from Fudan University, China, the M.Sc. degree in experimental medicine from McGill University, Canada, and the Ph.D. degree in computer science from Concordia University, Canada. He was a Post-Doctoral Fellow with Montreal Institute for Learning Algorithms (MILA), University of Montreal. He is currently a Professor with Beijing University of Posts and Telecommunications (BUPT). His research interests include multimodal representation learning and machine learning methods applied to healthcare.

Yue Yu is currently a Researcher with the Peng Cheng Laboratory, China; an Associate Professor with the College of Computer Science, National University of Defense Technology (NUDT); and a Technical Committee Member of the OpenI Community. His research has been published on ICLR, TSE, CHI, CSCW, ICSE, and ACL. His current research interests include software engineering and artificial Intelligence.

Wei-Shi Zheng is currently a Full Professor with Sun Yat-sen University. He has published more than 200 papers, including more than 150 publications in main journals, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *International Journal of Computer Vision*, and IEEE TRANSACTIONS ON IMAGE PROCESSING, and top conferences, such as ICCV, CVPR, SIGGRAPH, ECCV, and NeurIPS. His research interests include person/object association and activity understanding, and the related weakly supervised/unsupervised and continual learning machine learning algorithms. He was a recipient of the Excellent Young Scientists Fund of the National Natural Science Foundation of China and the Royal Society-Newton Advanced Fellowships, U.K. He has ever served as an area chair for ICCV, CVPR, ECCV, BMVC, and NeurIPS. He is currently an Associate Editors/on the Editorial Board of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *Artificial Intelligence*, and *Pattern Recognition*. He has ever joined Microsoft Research Asia Young Faculty Visiting Programme. He is a Cheung Kong Scholar Distinguished Professor.

Ruixuan Wang received the Ph.D. degree from the National University of Singapore in 2007. He was a Post-Doctoral Researcher with the University of Dundee, U.K. He is currently an Associate Professor with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China. His research interests include computer vision, medical image analysis, and machine learning.