Space-time Prompting for Video Class-incremental Learning

Yixuan Pei[1] Zhiwu Qing[2] Shiwei Zhang[3*] Xiang Wang[2]
Yingya Zhang[3] Deli Zhao[3] Xueming Qian[1,4*]

[1]Xi'an Jiaotong University
[2]Huazhong University of Science and Technology [3]Alibaba Group
[4]Shaanxi Yulan Jiuzhou Intelligent Optoelectronic Technology Co., Ltd
{peiyixuan@stu, qianxm@mail}.xjtu.edu.cn {qzw,wxiang}@hust.edu.cn
{zhangjin.zsw, yingya.zyy}@alibaba-inc.com zhaodeli@gmail.com

Abstract

Recently, prompt-based learning has made impressive progress on image class-incremental learning, but it still lacks sufficient exploration in the video domain. In this paper, we will fill this gap by learning multiple prompts based on a powerful image-language pre-trained model, i.e., CLIP, making it fit for video class-incremental learning (VCIL). For this purpose, we present a space-time prompting approach (ST-Prompt) which contains two kinds of prompts, i.e., task-specific prompts and task-agnostic prompts. The task-specific prompts are to address the catastrophic forgetting problem by learning multi-grained prompts, i.e., spatial prompts, temporal prompts and comprehensive prompts, for accurate task identification. The task-agnostic prompts maintain a globally-shared prompt pool, which can empower the pre-trained image models with temporal perception abilities by exchanging contexts between frames. By this means, ST-Prompt can transfer the plentiful knowledge in the image-language pre-trained models to the VCIL task with only a tiny set of prompts to be optimized. To evaluate ST-Prompt, we conduct extensive experiments on three standard benchmarks. The results show that ST-Prompt can significantly surpass the state-of-the-art VCIL methods, especially it gains 9.06% *on HMDB51 dataset under the* 1×25 *stage setting.*

1. Introduction

Video understanding [49, 11, 1, 32] has achieved outstanding performance with the quick development of deep neural networks. However, training such a model heavily depends on the currently available labeled data. When different classification tasks are presented sequentially in practical applications, a crucial problem is to maintain the accuracy of the model as data arrives over time. Under this circumstance, due to the limitations of memory and privacy, the model can just access the videos of the current task and few even no examples of historical tasks, which requires that the model should adapt to the current task without catastrophic forgetting [34]. The machine learning paradigm to handle the above challenges is called Video Class-Incremental Learning (VCIL) [35, 36, 61].

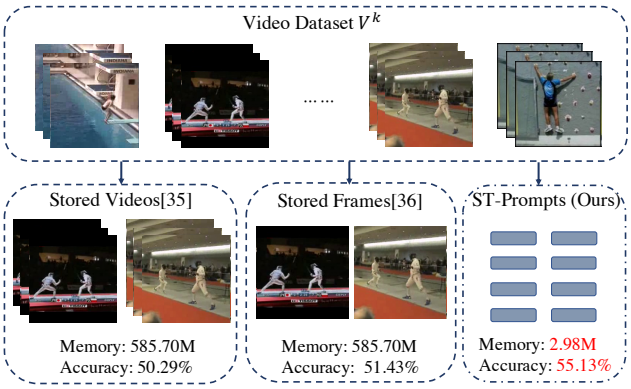

Figure 1. An comparison between existing VCIL methods based on exemplar or frame replay and our ST-Prompt. Compared to the rehearsal-based approaches, our ST-prompt can better solve the problem of catastrophic forgetting when only a small amount of memory is consumed to store a tiny set of prompts.

Most existing class-incremental learning approaches have achieved remarkable performance through a rehearsal buffer of the old tasks in both the image [40, 45, 59, 4] and video [36, 47, 36] fields. However, these methods suffer from suboptimal performance with smaller buffer sizes and severe catastrophic forgetting once the rehearsal buffer is not accessed, hence leading to limited applications. To solve this problem, several prompt-based learning methods in image domain [54, 53, 51] are proposed to train only

*Corresponding author.

a tiny set of parameters, *i.e*., prompts, to effectively instruct a pre-trained model to learn incremental tasks without buffering past examples. Typically, L2P [54] learns a single prompt pool to attach complementary prompts to the pre-trained backbone. Nevertheless, this new paradigm has shown great potential for images, but it is still underexplored in the video domain. In this paper, we will concentrate on exploring this paradigm on VCIL to bridge this gap. Directly applying these existing methods for the VCIL task tends to be inadequate. The core reason is that the prompt strategy for images cannot model temporal variations and dynamics across spatial features. Meanwhile, video data possesses a more complicated semantic structure than static images, so a single prompt pool like L2P may be insufficient to learn the abundant semantics. Therefore, we think a properly redesigned prompt framework will contribute to improving the performance of the VCIL task.

Motivated by the above observations, we propose ST-Prompt, a space-time prompting method for video class-incremental learning. Specifically, we design ST-Prompt with two components, *i.e*., *task-specific prompts* and *task-agnostic prompts*. The task-specific prompts are tapped by multi-grained prompts, *i.e*., spatial prompts, temporal prompts and comprehensive prompts, which can better discover the complicated spatio-temporal semantics in videos to address the catastrophic forgetting problem. The task-agnostic prompts learn a complementary prompt for each frame and then gather them back for all frames; hence we can equip the image pre-trained model with temporal modeling capacities without any edition of model structure. Compared with the task-specific prompts, the task-agnostic prompts are task-independent and thus share a generic and globally-shared prompt pool for all tasks. By this means, ST-Prompt has two advantages: 1) it can simultaneously take advantage of the knowledge priors in the image-language pre-trained model and learn the spatio-temporal information, which makes the model more suitable for VCIL; 2) it can markedly reduce the memory cost by just saving a set of prompt parameters without preserving redundant videos anymore, as shown in Figure 1. In the experimental stage, we quantificationally and qualitatively evaluate ST-Prompt on three standard benchmarks. The results show that ST-Prompt can achieve significant gains over the current top-performing approaches, which demonstrates the effectiveness of our method.

In summary, the main contributions of this paper are as follows:

- We first explore the great potential of image-language pre-trained model in video incremental tasks and achieve significant performance improvement by utilizing its sufficient semantic information through prompt-based methods;
- We proposed ST-Prompt, containing task-specific prompts and task-agnostic prompts, which can significantly reduce the memory cost and forgetting rate while modeling temporal variations and dynamics across spatial features without modifying the model structure;
- The proposed ST-Prompt achieves remarkable performance on multiple standard action recognition incremental benchmarks over state-of-the-art methods.

2. Related Work

In this section, we will introduce the methods related to ST-Prompt, including the works for image and video class-incremental learning, and prompt-based learning.

2.1. Image Class-Incremental Learning

In recent years, class-incremental learning has been widely studied and numerous methods have been proposed in the image domain [52, 56, 10, 43]. These methods can be roughly partitioned into three categories: regularization-based methods, exemplar-based methods and architecture-based methods. Regularization-based methods [23, 60, 21] constrain the optimization of network parameters to preserve the important information of learned tasks, in which the most popular technique utilized is knowledge distillation [29, 12, 14]. The exemplar-based methods [40, 57, 45] typically store a small set of representative exemplars [59, 50] or generated synthetic examples [22, 55] from previous classes to prevent catastrophic forgetting, which is always combined with regularization of a distillation loss to encourage knowledge retention further. The architecture-based methods [26, 42, 48] retain the knowledge of old tasks by designing specific components in the architecture [41, 64] or expanding current feature extractor for new data[58, 38].

2.2. Video Class-Incremental Learning

In the video domain, studies on this field are still scarce. The existing solutions are always the combination of regularization-based and exemplar-based methods and focus on how to formulate temporal information better. For example, TCD [35] computes time-channel importance for weighted distillation, [61] decomposes spatio-temporal knowledge transfer for stronger constraint and vCLIMB [47] focuses on temporal-consistency regularization of untrimmed video. Both TCD [35] and vCLIMB [47] have claimed that more stored examples in memory can effectively encourage incremental performance. Specifically, FrameMaker [36] tends to design a memory-efficient video incremental learning method by learning a condensed frame for each representative video of old classes. However, it still demands to preserve some old frames.

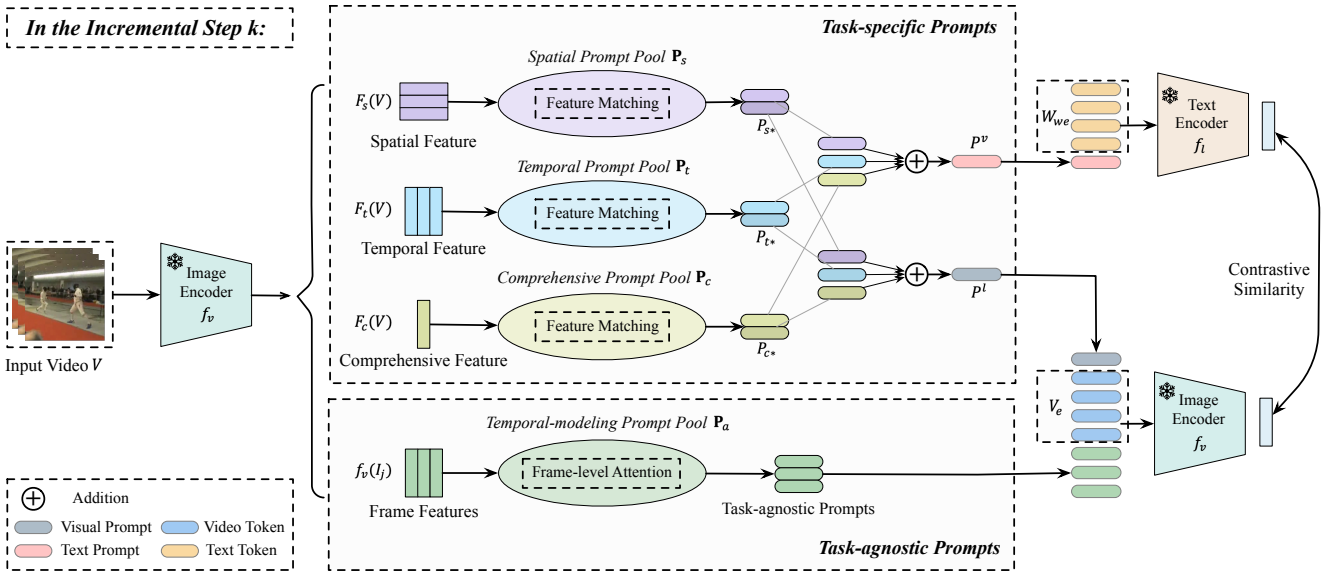

Figure 2. The overall framework of the proposed ST-Prompt, which mainly contains two kinds of prompts, *task-specific prompts* and *task-agnostic prompts*. For each input video, we first extract multi-grained features with the image encoder of a CLIP model, then select corresponding prompts from predefined prompt pools. Finally, the prompts are prepended to the video and text embedding feature respectively to instruct the model to learn incremental tasks.

2.3. Prompt-based Learning

Recently, with the rise of large pre-trained models [39, 3, 18, 28], the general pre-trained models are utilized to adapt to multiple down-sampling tasks by finetuning or prompting. The concept of prompt was first proposed in the field of natural language processing [31, 5, 37], and later it was more widely used in language[25, 27, 16] and visual[19, 2, 20] tasks to modify the input texts or images, such that the language or image model obtains additional information about a specific task. Some methods consider multi-modality information simultaneously, and design prompts for the vision-language models[63, 62, 17].

Meanwhile, the emergence of large pre-trained models and prompt-based learning also shows excellent potential in incremental learning. L2P [54], DualPrompt [53] and S- iPrompts [51] exploit prompting method to image incremental learning and achieve good performance without saving old exemplars. Especially, S-Prompting utilizes both image and text prompts together with a pre-trained vision-language model and gains outstanding performance in domain-incremental learning. In this paper, we focus on the properties of video, adapting large image pre-trained models to the video domain for video class-incremental learning by space-time prompting. We propose to decompose the complex video features into multiple space-time granularities for accurate task identification while exchanging the temporal information between frames by task-agnostic prompts.

3. Method

In this section, we will first introduce the basic setting of the video class-incremental learning task. Then we discuss the proposed ST-Prompt in detail, as shown in Figure 2. In the end, we present the training and inference procedure.

3.1. Problem Formulation

In video class-incremental learning, different tasks $\{\mathcal{T}^1, \mathcal{T}^2, \cdots, \mathcal{T}^K\}$ arrive in a sequence and each task $\mathcal{T}^k$ corresponds to its specific dataset $\mathcal{D}^k = \{(V_i^k, y_i^k), y_i^k \in L^k\}$, which contains the input video example V_i^k and its label y_i^k. The labels in task k belong to a predefined label set L^k and do not appear in previous tasks. The goal of VCIL is to train a single model that has the ability to classify videos of all seen classes correctly. We focus on the more challenging problem of exemplar-free video class-incremental learning, *i.e.*, no exemplar of old classes can be used in the current task, in which the core is to avoid forgetting historical information, termed *catastrophic forgetting*.

In this paper, we follow the prompt-based incremental learning paradigm in the image domain [54, 51] to alleviate the forgetting problem, which is composed of two stages: *i)* based on the pre-trained vision-language model like CLIP, they optimize a tiny set of learnable visual and text prompts for each incremental task independently (*i.e.* task-specific prompts) to learn new knowledge and retain the historical information; *ii)* during inference, due to the lack of task identity of the test example, they select the proper visual and text prompts from the prompt pool by prompt matching, and then prepend them to the visual embedding V_{ie}^k and the class

embedding W_{we}^k respectively. The final prediction is determined by the similarities between the visual features and the text features of class names. However, different from static images, videos contain more complicated spatio-temporal clues, and it is intractable to capture these semantic clues well in the video tasks via a single kind of prompt. Meanwhile, the image pre-trained model lacks temporal modeling capability for videos, which requires us to make up for this defect during the prompting process. Overall, the essential point for VCIL is to design a reasonable prompting method to recall the spatio-temporal information in the video task well.

3.2. Space-time prompting

The proposed space-time prompting (ST-Prompt) contains two components, *i.e.*, task-specific prompts and task-agnostic prompts.

Task-specific Prompts. To prevent catastrophic forgetting, we train task-specific prompts for each incremental task $\mathcal{T}^k$ with the pre-trained model (the image encoder f_v and text encoder f_l) frozen. Specifically, we devote to designing multi-grained prompts to utilize the temporal and spatial variations of different videos for more accurate task identification. Formally, for each task $\mathcal{T}^k$, there are three different granularities of task-specific prompt pools as Eq. 1. For brevity, we omit the symbol k in this section. If there is no additional explanation, it indicates the situation in incremental task $\mathcal{T}^k$.

$$\begin{aligned} &\mathbf{P} = \{\mathbf{P}_s, \mathbf{P}_t, \mathbf{P}_c\} \\ &\mathbf{P}_m = \{\{P_{m1}^v, P_{m1}^l\}, ..., \{P_{mN}^v, P_{mN_m}^l\}\}, \\ &P_{mn}^v \in \mathbb{R}^{L^v \times D^v}, P_{mn}^l \in \mathbb{R}^{L^l \times D^l}, m \in \{s, t, c\}, \end{aligned} \tag{1}$$

where $\mathbf{P}_s$, $\mathbf{P}_t$ and $\mathbf{P}_c$ are the spatial, temporal and comprehensive task-specific prompt pools. Each element in prompt pools is organized as the formulation of a visual-text prompt pair, where P_{mn}^v and P_{mn}^l are the commensurable visual and text prompt respectively. Additionally, N_m is the numbers of prompts in each prompt pool. L^v and L^l are the length of the visual and text prompt, and D^v and D^l are their embedding dimension.

The selection of prompts from these prompt pools is realized through feature matching as Figure 3(a). We apply K-Means algorithm to store the centroids of video features and search for the nearest centroid of the given video sample to identify its relevant prompt. The matching of three kinds of prompts corresponds to features in three different granularities of video V, *i.e.* spatial feature $F_s(V)$, temporal feature $F_t(V)$ and comprehensive feature $F_c(V)$. Specifically, the spatial and temporal features are calculated by pooling the temporal and spatial dimensions separately, while the comprehensive feature is the average of [CLS] token output. After the feature matching, the selected task-specific

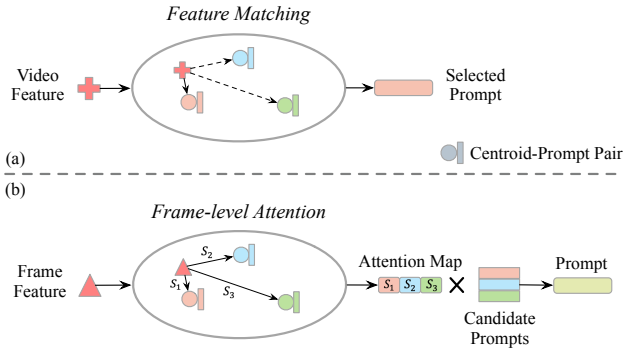

Figure 3. The details of prompt matching methods in prompt pools. (a) We select prompts corresponding to the nearest centroid with query feature for the task-specific prompts; (b) We utilize the similarities between the frame feature and centroids as the attention map for the task-agnostic prompts.

prompts are organized together as:

$$P^v = f(P_{s*}^v, P_{t*}^v, P_{c*}^v), P^l = f(P_{s*}^l, P_{t*}^l, P_{c*}^l) \tag{2}$$

where $(P_{s*}^v, P_{s*}^l), (P_{t*}^v, P_{t*}^l)$ and (P_{c*}^v, P_{c*}^l) are the prompts selected from three task-specific prompt pools. f is the organization function, which can be stacking, addition, multiplication, or other operations. In this paper, we use addition for its better performance.

Finally, the visual and text prompts are prepended to the visual and text embedding respectively, and the whole input of the visual and text encoder is of the format:

$$V_p = [P^v; V_e], W_{wp} = [P^l; W_{we}]. \tag{3}$$

where ; indicates the prepending operation, V_e is the video embedding feature of video V and W_{we} is the word embedding feature of w-th class name in task $\mathcal{T}^k$.

Additionally, when we utilize the visual pre-trained models without a language branch, such as ViT with ImageNet pre-trained weights, the classification head replaces the text encoder and the prompts in each prompt pool just contain the visual prompt P_{mn}^v. The other definition and utilization of the task-specific prompts are similar to the above.

Task-agnostic prompts. When tackling video incremental tasks, the image pre-trained model processes each video frame separately and cannot model the temporal variations and relations between different frames. Although we utilize different granularities of prompts for task identification, it is not enough to make the model understand the temporal information of videos well. To enhance the temporal modeling ability and adaptability of image pre-trained models, we propose task-agnostic prompts to highlight the different information contained in frames. Specifically, we establish an additional temporal-modeling prompt pool as $\mathbf{P_a} = \{P_{a1}, ..., P_{aN_a}\}$, where $P_{an} \in \mathbb{R}^{L^a \times D^v}$ is the prompt and N_a is the total number of prompts. Compared

with the task-specific prompts, temporal modeling is a more general concept for videos, which is task-agnostic and all prompts from the whole pool can be selected and attached without task limitation.

For the selection of task-agnostic prompts, we also store the centroids of the frame features during training by K-Means algorithm. Differently, in order to utilize the information in the temporal-modeling prompt pool comprehensively, the final prompt according to each frame is not selected by feature matching simply but is calculated by frame-level attention as Figure 3(b). For each frame I_j in video V, the attention map for all candidate prompts $S_j = \{S_{j1}, S_{j2}, ..., S_{jN_a}\}$ is generated according to the similarity between the frame feature $f_v(I_j)$ and the centroids. Specifically, we utilize the cosine similarity to calculate S_{jn} in this paper. And then the task-agnostic prompt corresponding to the frame I_j is calculated as follows:

$$P_j = \sum_{n=1}^{N_a} \frac{e^{S_{jn}/\tau}}{\sum_{n=1}^{N_a} e^{S_{jn}/\tau}} P_{an} \in \mathbb{R}^{D^v} \tag{4}$$

where τ is the temperature parameter. Finally, the prompts of different frames are stacked together to adapt to the video embedding as $V_p = [P_1; P_2; ...; P_T; V_e]$. In this way, each frame will obtain the context information of others. Although they go forward through an image pre-trained model separately, the information exchange of different frames realizes the temporal modeling.

3.3. Training and Inference

During training, only the prompts are trainable, while the pre-trained model is fixed. In incremental step $\mathcal{T}^k$, the training of prompts in task-specific prompt pools is restrained so that only the prompts corresponding to task $\mathcal{T}^k$ can be selected and utilized. The task-agnostic prompt pool does not have this constraint because temporal modeling is a more general concept independent of task identification.

During inference, for each new video, we first extract multi-grained features by the pre-trained image encoder f_v, and then select task-specific and task-agnostic prompts from prompt pools through feature matching or frame-level attention. Finally, the selected prompts are prepended to the video and text embedding features respectively. The classification result is according to the similarity output.

When ST-Prompt is utilized based on a one-branch visual model, in order to separate different tasks better, the classification head is also split according to the task identity and every incremental session trains its own classifier. Differently, without the text branch, the classifier cannot utilize the multi-grained task-specific prompts flexibly and can only be selected by the video-level comprehensive feature. The classifier selection corresponds to the task that the selected comprehensive task-specific prompt belongs to.

4. Experiment

In this section, we compare our results with baselines and state-of-the-art VCIL methods. We also show ablation studies on the effect of different components and analyze our results in different practical settings.

4.1. Experiment setup

Datasets. We evaluate our method on UCF101 [44], HMDB51 [24] and Something-Something V2 [13], which are the standard action recognition datasets. We follow the video frequently-used class-incremental benchmarks presented in TCD [35]. For UCF101, the model is trained on 51 classes first, and the other 50 classes are divided into 5, 10 and 25 tasks. For HMDB51, we train the base model using videos from 26 classes, and the rest of the classes are separated into 5 or 25 groups. For Something-Something V2, we first train 84 classes in the initial stage and generate the incremental groups of 10 and 5 classes.

Evaluation Protocol. After each incremental step, we evaluate the model on all seen classes and the class-incremental methods are finally assessed by the average accuracy of all tasks.

Implementation Details. Our method is based on CLIP ViT-B/16 [39], a visual-language model pre-trained with numerous image-text pairs. While the pre-trained model is frozen, the visual and textual prompts are optimized using a batch size of 64 and an initial learning rate of 0.001 with cosine descending for 50 epochs in each task. We carefully choose compared methods in the same environment for a fair comparison. However, most existing class-incremental learning methods are not adaptive to visual-language models. Therefore, we also apply our approach on ImageNet pre-trained ViT-B/16 [8] and reproduce other class-incremental learning methods on the same backbone.

4.2. Main Results.

We compare our proposed ST-Prompt with existing state-of-the-art class-incremental learning approaches under multiple challenging settings on three datasets in Table 1 and Table 2. For the approaches in the image domain, we reproduce them by treating the video as a whole as that in the image field. From the tables, we can draw the following conclusions. First, ST-Prompt notably outperforms existing rehearsal-free methods in VCIL, containing two recent prompt-based approaches, L2P and S-iPrompts in the image domain. With ImageNet pre-trained ViT, ST-Prompt respectively surpasses S-iPrompts by around 4.22%, 4.98% and 7.27% on HMDB51, UCF101 and Something-something v2, respectivly. This indicates that

Methods	Buffer Size	UCF101			HMDB51	
		10×5 stages	5×10 stages	2×25 stages	5×5 stages	1×25 stages
ImageNet pre-trained ViT-B/16						
iCaRL [40]	10/class	70.58	69.51	67.28	43.90	37.15
UCIR [15]		77.55	74.59	71.77	48.20	39.42
PODNet [9]		76.50	76.17	73.83	48.38	43.35
Co^{2}L [6]		78.54	77.13	75.59	49.76	44.15
TCD† [35]		78.13	76.87	75.74	50.29	44.04
FrameMaker† [36]		79.37	79.55	79.32	51.43	46.37
L2P [54]		81.24	80.09	78.58	49.98	45.87
iCaRL [40]	5/class	66.58	67.70	64.27	41.08	36.13
UCIR [15]		75.55	72.78	66.76	45.38	38.40
PODNet [9]		74.50	74.36	72.82	45.56	42.33
Co^{2}L [6]		77.57	76.32	74.58	47.10	43.56
TCD† [35]		76.13	75.06	73.73	48.38	43.02
FrameMaker† [36]		78.37	77.74	76.31	50.58	45.65
L2P [54]		79.93	77.78	76.27	47.67	42.56
Finetuning	0/class	24.83	13.61	6.24	18.30	4.89
Finetuning-frozen		67.10	65.71	7.39	36.64	4.95
LwFMC [29]		45.73	28.01	16.43	27.01	17.21
LwM [7]		46.98	28.70	16.92	27.94	17.85
L2P [54]		76.72	75.11	73.26	45.00	39.89
S-iPrompts [51]		77.16	77.83	77.99	50.67	51.45
ST-Prompt†		**83.74**	**81.53**	**82.66**	**55.13**	**55.43**
CLIP ViT-B/16						
Zero shot [39]	0/class	69.68	69.61	69.87	40.19	40.42
S-liPrompts [51]		80.60	80.27	80.43	53.11	53.89
ST-Prompt†		**84.75**	**85.54**	**85.67**	**60.14**	**60.54**

Table 1. Comparison with the state-of-the-art approaches over class-incremental action recognition performance on UCF101 and HMDB51. Compared methods are grouped based on different rehearsal buffer sizes. † indicates the methods designed for video.

our design of multi-grained task-specific prompts and task-agnostic prompts plays an essential role in video feature formulation and temporal perception. The more tremendous increase on Ssv2 also claims the effectiveness of ST-Prompt for the motion-sensitive dataset. Second, compared with rehearsal-based methods, our ST-Prompt also has a significant advantage. ST-Prompt achieves better performance than the top existing approach, FrameMaker, by approximately 6.38% and 3.23% on HMDB51 and UCF101, separately, although FrameMaker stores additional 80 examples for each class. Third, using the pre-trained vision-language model, CLIP, as the backbone for VCIL, the performance of ST-Prompt improves further by about 5.06%, 2.67% and 3.49% respectively. The reason is that CIIP provides more abundant prior knowledge about semantic information, which is vital for subsequent prompting.

4.3. Ablation Study.

In this section, we present ablation studies on the properties and effectiveness of our core designs. If not specified, the ablation studies are performed on HMDB51 with 5 steps and UCF101 with 10 steps.

Task-specific Prompts. In Table 3, we first explore how to organize different kinds of task-specific prompts together in Eq. 2. The results show that the addition operation achieves better performance than stacking and multiplication, whose reason may be that different prompts can modulate each other better without prepending too many parameters to the embedding feature in this way.

Then, to prove the effectiveness of different parts of our task-specific prompts, we remove each kind of prompts respectively under the combination method of addition and show their results in Table 4. From the results, we can find that: first, when three kinds of prompts are utilized separately, the comprehensive task-specific prompts achieve the best performance, which demonstrates that they depict the video feature more integrally; second, when the comprehensive prompts are combined with the temporal or spatial prompts, the performance increases by 1.9% or 1.54% on HMDB51, and 1.26% or 1.34% on UCF101, respectively. Differentiated promotions in different datasets imply that

Methods	Buffer Size	10×9 Stages	5×18 Stages
ImageNet pre-trained ViT-B/16			
iCaRL [40]	5/class	20.41	16.62
UCIR [15]		24.32	19.31
PODNet [9]		27.63	20.14
TCD† [35]		29.32	24.69
FrameMaker† [36]		31.41	26.57
L2P [54]		26.02	21.33
Finetuning	0/class	11.04	6.23
Finetuning-frozen		7.99	5.97
LwFMC [29]		15.21	11.41
L2P [54]		23.25	18.49
S-iPrompts [51]		30.58	26.33
ST-Prompt		**36.84**	**31.60**
CLIP ViT-B/16			
Zero-shot	0/class	3.56	3.57
S-liPrompts [51]		33.69	30.84
ST-Prompt†		**39.98**	**35.44**

Table 2. Comparison with the top approaches on Something-something v2.† indicates the methods designed for video.

Operation	Stacking	Multiplication	Addition
HMDB51	55.24	54.96	**56.02**
UCF101	82.82	82.45	**83.58**

Table 3. Ablation for the ensemble ways of task-specific prompts.

STP	TTP	CTP	HMDB51	UCF101
✓	✗	✗	44.62	72.45
✗	✓	✗	45.46	73.84
✗	✗	✓	53.11	80.27
✓	✓	✗	49.34	76.93
✓	✗	✓	54.65	81.61
✗	✓	✓	55.01	81.53
✓	✓	✓	**56.02**	**83.58**

Table 4. Ablations for task-specific prompts on HMDB51 with 5 steps and UCF101 with 10 steps. STP, TTP and CTP represent spatial, temporal and comprehensive prompts respectively.

temporal information is more critical for HMDB51, while the spatial feature for UCF101; third, utilizing all three kinds of task-specific prompts can further improve the final accuracy, which proves the combination of multi-grained prompts contribute to more correct task identification.

Further, we visualize some clustering results of different granularities of the video feature in Figure 4. The comprehensive feature judges video task-id from the overall perspective, and the clustered videos usually belong to the same spatial and temporal clusters, such as "Smile". The temporal and spatial prompts implement the local information for more complicated video task matching. Videos

Method	TAP	HMDB51	UCF101
TSP	✗	56.02	83.58
TSP+LE	✗	56.06	83.55
TSP+SP	✗	57.04	83.87
TSP+CP	✗	58.30	84.05
TSP+PE [46]	✗	57.96	84.04
TSP+SH [33]	✗	58.27	84.25
TSP+TS [30]	✗	58.74	84.34
ST-Prompt	✓	**60.14**	**85.54**

Table 5. Comparison with some baseline methods about task-agnostic prompts (TAP) on HMDB51 with 5 steps and UCF101 with 10 steps.

Method	TAP	HMDB51	UCF101
L2P [54]	✗	45.00	75.11
	✓	46.76	76.07
S-iPrompts [51]	✗	50.67	77.83
	✓	54.44	79.74
S-liPrompts [51]	✗	53.11	80.27
	✓	55.77	81.82
ST-Prompt	✗	56.02	83.58
	✓	**60.14**	**85.54**

Table 6. Ablations for task-agnostic prompts (TAP) on HMDB51 with 5 steps and UCF101 with 10 steps.

with the same motion but different spatial features are gathered together according to their temporal features, such as "Dive" in different scenarios, while spatial features distinguishes actions with mixed temporal information but obvious spatial similarities.

Task-agnostic Prompts. We compare our task-agnostic prompts for temporal modeling with a set of baselines in the proposed framework in Table 5. "LE" represents prompt length expansion, which excludes the performance improvement caused by increasing the number of parameters. And the results also prove that learning task-agnostic prompts about spatial ("SP") or comprehensive ("CP") information are not as effective as temporal modeling. Then we compare the effect of our temporal-modeling prompts with some other parameter-free temporal modeling methods, such as positional encoding ("PE"), temporal shuffle ("SH") and temporal shift ("TS"). We can observe that utilizing task-agnostic prompts to exchange the context of different frames can gain the best performance of 60.14%. Further, we validate the effectiveness of task-agnostic prompts by adding them to other prompt-based methods in the image domain, as shown in Table 6, which indicates that it is a general approach to improve the ability of temporal perception for image pre-trained models.

In addition, We depict the change of category-wise ac-

Methods	Backbone	Stored Exemplars	Memory Cost	Average Accuracy
TCD [35]	ImageNet Pre-trained ViT-B/16	10 examples/class	585.70M	50.29
FrameMaker [36]	ImageNet Pre-trained ViT-B/16	80 examples/class	585.70M	51.43
ST-Prompt	ImageNet Pre-trained ViT-B/16	0	2.98M	55.13
ST-Prompt	CLIP ViT-B/16	0	4.47M	60.14

Table 7. The comparison of the overhead of our ST-Prompt and other VCIL methods on HMDB51 with 5 steps.

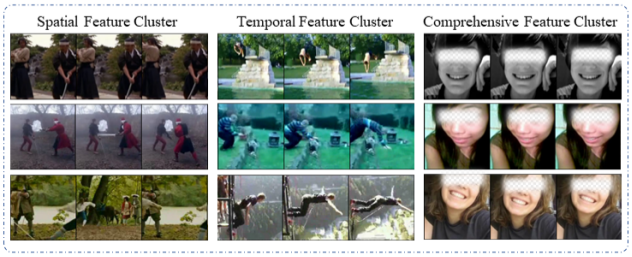

Figure 4. The visualization of multi-grained video feature clustering results.

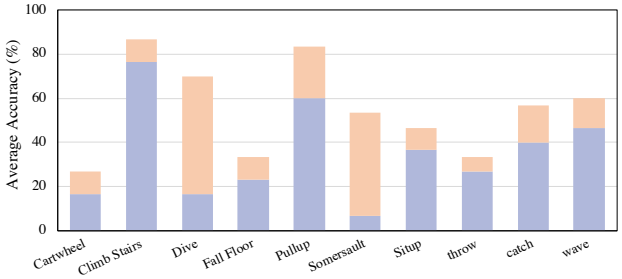

Figure 5. The change of category-wise accuracy after utilizing task-agnostic prompts on HMDB51 with 5 steps.

curacy after utilizing task-agnostic prompts in Figure 5. The significant increase in the accuracy of some motion-sensitive classes, such as "Dive" and "Somersault", proves the role of our task-agnostic prompts in improving the temporal modeling capability of models.

The Temperature τ. The temperature τ in Eq. 4 decides the smoothness of the attention weights. Too small τ makes the final prompt of each frame lose the understanding of global information, and too large τ introduces noise. From Figure 6(a), we observe that $\tau = 0.3$ guides the best performance of final average accuracy.

The Length of Prompts. The length of different kinds of prompts L^v, L^t, L^a decides the capacity of each learnable prompt. Figure 6(b) illustrates how the average accuracy changes when the length of different prompts varies. It is evident that a too-small visual or text prompt length always negatively affects results, while an oversized prompt may cause underfitting. We set the prompt length as 5 for both visual and text task-specific prompts according to the curves. In addition, we can discover that the final accuracy is not sensitive to the length of task-agnostic prompts

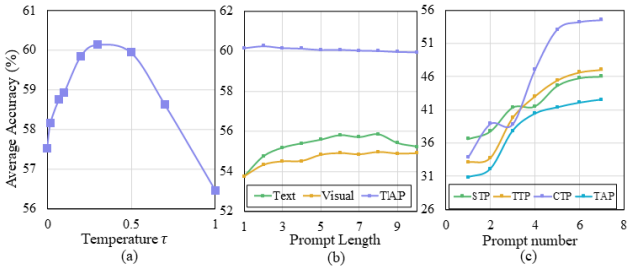

Figure 6. The effects of some hyperparameters in ST-Prompt.

and even decreases during the length is larger. Hence, we choose the length of the task-agnostic prompts as 1.

The Number of Prompts. Figure 6(c) presents how the number of different prompts $N_m (m \in \{s, t, c, a\})$ in each incremental task influences the performance of our approach, where the number in the base task is calculated according to the proportion of the number of classes. Three different task-specific prompts and task-agnostic prompts are explored separately. In general, the more prompts per task, *i.e.*, the more pre-defined centroids per task during clustering, the higher accuracy gained. Especially, the comprehensive prompt is more sensitive to the cluster number. For simplicity, the number of different prompts is set as 5.

4.4. Overhead Analysis

We compare the memory cost and average accuracy of ST-Prompt with other VCIL methods in Table 7. Our ST-Prompt utilizes only a set of prompts and prompt keys in each task, while previous rehearsal-based methods store a number of video examples or frames. Therefore, ST-Prompt has significant memory advantages over previous VCIL methods. Our ST-Prompt achieves performance increasing by around 8.71% with only 0.76% memory cost of other methods.

5. Discussions

Limitation. Compared with the previous video incremental learning approaches, ST-Prompt achieves a significant performance boost without any additional memory cost for video example saving. Nevertheless, ST-Prompt relies on the large pre-trained model, which may fail when the knowledge contained in it is not sufficient. Meanwhile, our

ST-Prompt prepends prompts to the embedding feature and has not discussed the phenomenon that prompts are prefixed to the hidden layers.

Conclusion. In this paper, we propose ST-Prompt, a novel space-time prompting framework for video class-incremental learning. It explores how to address the VCIL problem in a rehearsal-free way by learning multiple prompts based on a powerful image-language pre-trained model. ST-Prompt mainly consists of two components, task-specific prompts and task-agnostic prompts. The task-specific prompts tackle the catastrophic forgetting problem by learning multi-grained prompts for task identification. The task-agnostic prompts implement the temporal modeling ability of the image pre-trained model. In this way, ST-Prompt achieves a new state-of-the-art performance, which has already surpassed the methods with a memory buffer. We hope this simple and practical framework can inspire attention to prompting-based methods in VCIL.

Acknowledgments. This work was supported in part by the NSFC under Grant 62272380 and 62103317, the Science and Technology Program of Xi'an, China under Grant 21RGZN0017, and by Alibaba Group through Alibaba Innovative Research Program.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *ICCV*, 2021. 1

[2] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Visual prompting: Modifying pixel space to adapt pre-trained models. *arXiv preprint arXiv:2203.17274*, 2022. 3

[3] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. In *ICLR*, 2022. 3

[4] Zalán Borsos, Mojmir Mutny, and Andreas Krause. Coresets via bilevel optimization for continual learning and streaming. *Advances in Neural Information Processing Systems*, 2020. 1

[5] 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. In *NeurIPS*, 2020. 3

[6] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2l: Contrastive continual learning. In *ICCV*, 2021. 6

[7] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *CVPR*, 2019. 6

[8] 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. In *ICLR*, 2021. 5

[9] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *ECCV*, 2020. 6, 7

[10] Arthur Douillard, Alexandre Ramé, Guillaume Couairon, and Matthieu Cord. Dytox: Transformers for continual learning with dynamic token expansion. In *CVPR*, 2022. 2

[11] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 1

[12] Tao Feng, Mang Wang, and Hangjie Yuan. Overcoming catastrophic forgetting in incremental object detection via elastic response distillation. In *CVPR*, 2022. 2

[13] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017. 5

[14] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Lifelong learning via progressive distillation and retrospection. In *ECCV*, 2018. 2

[15] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *CVPR*, 2019. 6, 7

[16] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *ICML*, 2019. 3

[17] Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022. 3

[18] 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 *ICML*, 2021. 3

[19] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022. 3

[20] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. In *ECCV*, 2022. 3

[21] Minsoo Kang, Jaeyoo Park, and Bohyung Han. Class-incremental learning by knowledge distillation with adaptive feature consolidation. In *CVPR*, 2022. 2

[22] Ronald Kemker and Christopher Kanan. Fearnet: Brain-inspired model for incremental learning. In *ICLR*, 2018. 2

[23] 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*, 2017. 2

[24] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, 2011. 5

[25] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *EMNLP*, 2021. 3

[26] Xilai Li, Yingbo Zhou, Tianfu Wu, Richard Socher, and Caiming Xiong. Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. In *ICML*, 2019. 2

[27] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL-IJCNLP*, 2021. 3

[28] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. In *ICLR*, 2022. 3

[29] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 2017. 2, 6, 7

[30] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 2019. 7

[31] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 2022. 3

[32] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, 2022. 1

[33] Pingchuan Ma, Yao Zhou, Yu Lu, and Wei Zhang. Learning efficient video representation with video shuffle networks. *arXiv preprint arXiv:1911.11319*, 2019. 7

[34] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*. 1989. 1

[35] Jaeyoo Park, Minsoo Kang, and Bohyung Han. Class-incremental learning for action recognition in videos. In *ICCV*, 2021. 1, 2, 5, 6, 7, 8

[36] Yixuan Pei, Zhiwu Qing, Jun CEN, Xiang Wang, Shiwei Zhang, Yaxiong Wang, Mingqian Tang, Nong Sang, and Xueming Qian. Learning a condensed frame for memory-efficient video class-incremental learning. In *NeurIPS*, 2022. 1, 2, 6, 7, 8

[37] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? In *EMNLP*, 2019. 3

[38] Qi Qin, Wenpeng Hu, Han Peng, Dongyan Zhao, and Bing Liu. Bns: Building network structures dynamically for continual learning. *Advances in Neural Information Processing Systems*, 2021. 2

[39] 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 *ICML*, 2021. 3, 5, 6

[40] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *CVPR*, 2017. 1, 2, 6, 7

[41] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 2

[42] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *ICML*, 2018. 2

[43] Yujun Shi, Kuangqi Zhou, Jian Liang, Zihang Jiang, Jiashi Feng, Philip HS Torr, Song Bai, and Vincent YF Tan. Mimicking the oracle: An initial phase decorrelation approach for class incremental learning. In *CVPR*, 2022. 2

[44] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 5

[45] Rishabh Tiwari, Krishnateja Killamsetty, Rishabh Iyer, and Pradeep Shenoy. Gcr: Gradient coreset based replay buffer selection for continual learning. In *CVPR*, 2022. 1, 2

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 2017. 7

[47] Andrés Villa, Kumail Alhamoud, Victor Escorcia, Fabian Caba, Juan León Alcázar, and Bernard Ghanem. vclimb: A novel video class incremental learning benchmark. In *CVPR*, 2022. 1, 2

[48] Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Foster: Feature boosting and compression for class-incremental learning. In *ECCV*, 2022. 2

[49] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 1

[50] Liyuan Wang, Xingxing Zhang, Kuo Yang, Longhui Yu, Chongxuan Li, Lanqing Hong, Shifeng Zhang, Zhenguo Li, Yi Zhong, and Jun Zhu. Memory replay with data compression for continual learning. In *ICLR*, 2022. 2

[51] Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. S-prompts learning with pre-trained transformers: An occam's razor for domain incremental learning. In *NeurIPS*, 2022. 1, 3, 6, 7

[52] Zhen Wang, Liu Liu, Yiqun Duan, Yajing Kong, and Dacheng Tao. Continual learning with lifelong vision transformer. In *CVPR*, 2022. 2

[53] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *ECCV*, 2022. 1, 3

[54] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *CVPR*, 2022. 1, 2, 3, 6, 7

[55] Chenshen Wu, Luis Herranz, Xialei Liu, Joost van de Weijer, Bogdan Raducanu, et al. Memory replay gans: Learning to generate new categories without forgetting. In *NeurIPS*, 2018. 2

[56] Tz-Ying Wu, Gurumurthy Swaminathan, Zhizhong Li, Avinash Ravichandran, Nuno Vasconcelos, Rahul Bhotika, and Stefano Soatto. Class-incremental learning with strong pre-trained models. In *CVPR*, 2022. 2

[57] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *CVPR*, 2019. 2

[58] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *CVPR*, 2021. 2

[59] Jaehong Yoon, Divyam Madaan, Eunho Yang, and Sung Ju Hwang. Online coreset selection for rehearsal-based continual learning. *arXiv preprint arXiv:2106.01085*, 2021. 1, 2

[60] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *ICML*, 2017. 2

[61] Hanbin Zhao, Xin Qin, Shihao Su, Yongjian Fu, Zibo Lin, and Xi Li. When video classification meets incremental classes. In *ACM MM*, 2021. 1, 2

[62] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, 2022. 3

[63] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 2022. 3

[64] Kai Zhu, Wei Zhai, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Self-sustaining representation expansion for non-exemplar class-incremental learning. In *CVPR*, 2022. 2