ViLT-CLIP: Video and Language Tuning CLIP with Multimodal Prompt Learning and Scenario-Guided Optimization

Hao Wang, Fang Liu*, Licheng Jiao, Jiahao Wang, Zehua Hao, Shuo Li
Lingling Li, Puhua Chen, Xu Liu

Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education,
Joint International Research Laboratory of Intelligent Perception and Computation,
School of Artificial Intelligent, Xidian University, Xi'an, 710071, P.R. China
stuhaowang@163.com, f63liu@163.com, lchjiao@mail.xidian.edu.cn

Abstract

Pre-trained vision-language(V-L) models such as CLIP have demonstrated impressive Zero-Shot performance in many downstream tasks. Since adopting contrastive video-text pairs methods like CLIP to video tasks is limited by its high cost and scale, recent approaches focus on efficiently transferring the image-based CLIP to the video domain. A major finding is that fine-tuning the pre-trained model to achieve strong fully supervised performance leads to low zero-shot, few-shot, and base to novel generalization. Instead, freezing the backbone network to maintain generalization ability weakens fully supervised performance. Otherwise, no single prompt tuning branch consistently performs optimally. In this work, we proposed a multimodal prompt learning scheme that balances supervised and generalized performance. Our prompting approach contains three sections: 1) Independent prompt on both the vision and text branches to learn the language and visual contexts. 2) Inter-modal prompt mapping to ensure mutual synergy. 3) Reducing the discrepancy between the hand-crafted prompt (`a video of a person doing [CLS]`) and the learnable prompt to alleviate the forgetting about essential video scenarios. Extensive validation of fully supervised, zero-shot, few-shot, base-to-novel generalization settings for video recognition indicates that the proposed approach achieves competitive performance with less commute cost.

Introduction

Vision language pre-training models like CLIP (Radford et al. 2021) and ALIGN (Jia et al. 2021) are trained using millions of image-text pairs crawled from the Internet without labels. The models can learn semantic representations from large amounts of unlabeled data while having demonstrated excellent "zero-shot" generalization for various image classification tasks (Li et al. 2023). However, adopting such massive training to video-level tasks faces the following difficulties: **1)** The availability of equivalent video-text datasets is comparatively limited, and preparing such video-text pairs data with complex annotation requires drives a monumental cost in contrast to image-text pairs (Jia et al. 2021). **2)** Solving video tasks demands more computational power. Given the same budget, training on image-text pairs enables the model to learn more. **3)** The large noise in videos makes video-text pertaining challenging to play a significant role. How to transfer the capabilities of large-scale pre-trained models and effectively adapt them to video-based tasks is a crucial research question.

It is essential for advancing video understanding and unlocking the potential of pre-trained models in video tasks. However, recent studies in this domain encounter a trade-off. Freezing encoders to maintain zero-shot capability will reduce the downstream supervised accuracy. Adapting image encoders to different downstream tasks by fine-tuning them end-to-end requires the development of numerous models for each task, which yields significant resource requirements. Fine-tuning the pre-trained model to achieve strong supervised performance causes low zero-shot generalization. Moreover, fine-tuning the pre-trained model to achieve strong supervised performance results or dumping the text encoder will lose the "Zero-Shot" generalization ability of the pre-training model. In contrast, CLIP provides a more versatile approach by utilizing relevant "prompts" for adapting to downstream tasks. This method eliminates the need for domain experts with extensive expertise, as it can be applied to various domains and tasks by learning to formulate and adjust solutions.

Therefore, recent video-based approaches typically freeze the encoder and adopt the CLIP representations along with additional learnable components. These components include a transformer-based temporal module (Ju et al. 2022), new cross-frame communication attention for video temporal modeling and a video-specific prompting technique (Ni et al. 2022), textual or visual prompts (Wang et al. 2021), a spatial adaptation, temporal adaptation, and joint adaptation module (Yang et al. 2023), that are learned while keeping the CLIP backbone frozen or adapting the CLIP encoders as well. These are designed to adapt CLIP while learning them quickly. However, these designs tend to drop CLIP's original generalization capability (Rasheed et al. 2023). Moreover, adapting representations in a single branch of CLIP is sub-optimal since stands difficult to flexibly and dynamically adjust both representation spaces on video tasks and alignment between the vision and language representations is lacking (Khattak et al. 2023).

To overcome the aforementioned challenges, we propose a video and language tuning CLIP with multi-modal prompt

*Corresponding author

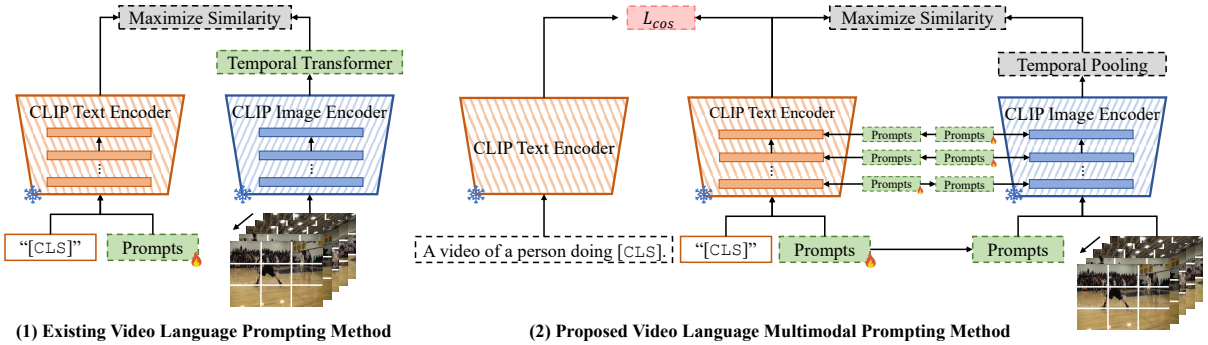

Figure 1: Comparison of ViLT-CLIP with existing prompt tuning methods. 1) Existing methods adapt textual prompting techniques to fine-tune the CLIP representation while freezing the visual branch and feeding the visual representations into a transformer layer for temporal modeling. 2) ViLT-CLIP introduces multi-modal branch-aware hierarchical prompts that adapt both language and vision branches with a Scenario-guided optimizer branch to improve generalization.

learning. Transferring the pre-trained image-text model to the video domain has two key considerations: **1)** It is essential to freeze the encoder parameters to preserve the original abilities. **2)** We recommend lightweight prompt learning on the visual side to adapt to the video domain. Our approach consists of three modules: **1)** Branch-aware hierarchical prompts are introduced, incorporating learnable context at each layer of the visual encoder. **2)** A cross-bidirectional linear layer is used to share two modal prompts, ensuring synergy between different branches and achieving completeness of contextual information. **3)** Additionally, we notice that the aligned video datasets used in classification tasks have quite limited textual information. We are limited to relying on class labels as textual descriptions instead of having per-sample textual descriptions such as (Radford et al. 2021). Inspired by (Zhou et al. 2022b), we introduce a prompt approach incorporating learnable context at each layer of the text side encoder. It can effectively model textual context and replenish the video category description. Simultaneously, building upon insights from (Yao, Zhang, and Xu 2023), we introduce a constrained branch that aims to reduce the discrepancy between the text embeddings of learnable prompts and the hand-crafted template prompt (`a video of a person doing [CLS].`) to relieve the forgetting about essential knowledge in the video domain. An overview of our method is presented in Figure 1. The contributions of this work are as follows,

- We propose a Video and Language tuning CLIP with multi-modal prompt learning and scenario-guided for adapting image-based CLIP to video-specific tasks, including video recognition and video text retrieval.
- Three types of prompt learning methods using a frozen backbone. On the vision branch, per-layer deep prompts are introduced to bridge the modality gap in videos. On the language branch, our approach efficiently learns semantic context and alleviates the forgetting of essential knowledge in text descriptions.
- We extensively evaluate the effectiveness of the ViLT-CLIP approach on public benchmarks such as video classification in supervised, few-shot, zero-shot, base-to-novel generalization settings, and video retrieval. Our approach bridges the modality gap through multi-modal prompt tuning with the frozen CLIP model.

Related Work

Vision Language Models

Vision-language (V-L) models (Radford et al. 2021; Jia et al. 2021; Yao et al. 2021) have recently garnered significant attention in the computer vision community. These V-L models aim to learn a joint image-language representation space using abundantly available image-text pairs from the web and achieve exceptional generalization performance. This ability has been transferable to many downstream tasks: semantic segmentation (Li et al. 2022a; Rao et al. 2022), object detection (Gu et al. 2021) and video classification (Xu et al. 2021). In the video domain, there exist several approaches trained with video-text pairs for applications such as video retrieval (Lei et al. 2021; Ni et al. 2022). However, it is worth noting that these models are not trained on large-scale video-text data, which may limit their performance and generalization capabilities in complex video tasks. In this work, we propose a novel approach to induce temporal cues with a multi-modal prompt learning technique to effectively adapt CLIP for supervised and generalized video recognition tasks and video text retrieval tasks.

Prompt Learning

As a recently adapted paradigm in NLP, text prompt is usually used in the language branch of the V-L model. It is a template in the form of a sentence, which can be handcrafted for a downstream task or autonomously learned during the fine-tuning stage. The latter is known as "Prompt Learning", which efficiently adapts a language model to downstream tasks by utilizing a few additional learnable tokens at the input. CoOp (Zhou et al. 2022b) and CoCoOp (Zhou et al. 2022a) propose to use sets of learnable prompts along with input to adapt the language representation. MaPLe (Khattak et al. 2023) proposes multi-modal prompting to improve alignment between the vision and

language representations. Similarly, Visual Prompt Tuning (VPT) (Jia et al. 2022) has analyzed how to embed learnable tokens on the vision branch. In a video task, Ju et al. (Ju et al. 2022) introduce text prompts and transformer layers for temporal modeling to efficiently adapt CLIP.

Video Action Recognition

Video recognition research focuses on how to construct an efficient temporal model. Over many years of research, many methods have been discovered. Among them, optical flow is widely used for two-stream fusion methods (Simonyan and Zisserman 2014; Wang et al. 2016), but optical flow calculation is time-consuming and storage-intensive. 3D convolution extends the traditional 2D convolution in the image space domain to the space-time domain (Tran et al. 2015; Carreira and Zisserman 2017; Tran et al. 2018), but it has poor targeting and high computational complexity. Other studies focus on inserting a plug-and-play temporal module into 2D convolutional networks (Lin, Gan, and Han 2019; Liu et al. 2021) or transferring LSTM's sequence data processing capability. Recently, network structures based on Vision Transformer (Bertasius, Wang, and Torresani 2021; Fan et al. 2021; Liu et al. 2022c; Li et al. 2022b) have shown more efficient temporal modeling and feature extraction capabilities. Except for single-modality methods, some multimodal alignment methods (Wang et al. 2021; Luo et al. 2022; Ju et al. 2022; Ni et al. 2022) transfer the CLIP's ability to video classification and video text retrieval tasks.

Video Text Retrieval

Early work (Liu et al. 2022b; Gabeur et al. 2020) mainly focused on fusing multimodal features to improve retrieval performance. Later work (Chen et al. 2020; Wu et al. 2021) considered decomposing cross-modal video text retrieval into matching between three levels (events, actions, entities) from global to local. Inspired by the concept of large-scale pre-training models (such as BERT (Devlin et al. 2018), CLIP, etc.). Recent works (Luo et al. 2022) discover a common video-text joint representation space by contrastive learning (Liu et al. 2022a) or masked language modeling. Our work suggests directly adapting the frozen CLIP by learnable prompt inputs, transferring its flexible capabilities to video retrieval downstream. Empirical studies demonstrate that our method is effective.

ViLT-CLIP: Methodology

Our proposed method aims to migrate CLIP capabilities to downstream video tasks through multimodal V-L prompting. Figure 2 shows the overall architecture of our method ViLT (Video Language Tuning with Multimodal prompt learning). ViLT learns a joint representation of prompts in the vision and language branch of CLIP. Specifically, we introduce a set of learnable vectors in the shallow layers of the CLIP language branch, which cooperate with the visual prompt vector on the shallow layer of the visual branch through a tiny linear network. At the same time, the embedded prompt vector on the deep layer of the language branch is associated with the deep layer of the visual branch in the same way. To learn the context representation of hierarchy, we apply deep prompt tuning in all transformer layers of both branches. Below, we will introduce deep language prompt tuning, deep visual prompt tuning, and our multimodal V-L prompt tuning in detail.

Video and Text Encoder

Text Encoder The pretrained text encoder is a 12-layer transformer model with an embedding size of 512 and a context length of 77. These layers utilize Multi-Head Self Attention (MHSA) followed by a Feed-Forward Network (FFN) to capture word relationships and contextual information within the text sequences. Given category text descriptions $C = \{c_0, c_1, \cdots c_{k-1}\}$ for a video, k denotes the number of categories, we generally construct a sentence using a hand-crafted prompt for the text description like "`A video of the action of {[CLS]}`". Then we use the text encoder $\mathcal{T}$ to obtain per-category representation $\mathbf{c} = \{\mathcal{T}(c_1), \mathcal{T}(c_2), \cdots, \mathcal{T}(c_{k-1})\}$. Inspired by (Zhou et al. 2022b,a; Yao, Zhang, and Xu 2023), we replace the hand-crafted prompt with the learnable prompt to tune the text branch. Furthermore, we also preserve the prior knowledge in the hand-crafted prompt.

Video Encoder The ImageEncoer in CLIP is based on the Vision Transformer (ViT) (Dosovitskiy et al. 2021) architecture. For a video data $V \in \mathbb{R}^{T\times 3\times H\times W}$ with T frames and spatial size $H \times W$. Each video frame $\{I_t\}_{t=1}^{T}$ is split into $M = H \times W/P^2$ fixed-size patches of size $P \times P$ and which are flattened into a set of vectors $\{\mathbf{x}_{t,i} \in \mathbb{R}^{P^2}\}_{i=1}^{M}$, where t denotes the frame number and i the patch number. Then these vectors are projected into patch embedding $\mathbf{E}_{t,0} \in \mathbb{R}^{M\times d_v}$ via a linear projection $\boldsymbol{p}$, with the dimension of tokens being d_v. $\mathbf{E}_{t,i}$ are then input to the $(i+1)^{th}$ transformer layer along with an additional embedded class (CLS) token, $\mathbf{x}^{cls} \in \mathbb{R}^{d_v}$, is independent for each frame:

$$\mathbf{E}_{t,\,0} = [\mathbf{x}_{t,0}^{cls}, \boldsymbol{p}^T\mathbf{x}_{t,1}, \boldsymbol{p}^T\mathbf{x}_{t,2}, \cdots, \boldsymbol{p}^T\mathbf{x}_{t,M}] + \mathbf{e}_0, \quad (1)$$

$$[\mathbf{E}_{t,\,i}] = \mathcal{V}_i([\mathbf{E}_{t,\,i-1}]) \quad i = 1, 2, \cdots, N, \quad (2)$$

where $\mathbf{e}_0$ denote the positional encoding, $\mathcal{V}_i$ the i-th layer of the video encoder $\mathcal{V}$. Finally, we obtain the frame level features at each layer and the final classification representation $\mathbf{x}$, the class token $\mathbf{x}_{t,N}^{cls}$ of the last transformer layer output $\mathbf{E}_{t,N}$ is projected to a common V-L embedding space via `ImageProj` as follows:

$$\mathbf{z}_t = \texttt{ImageProj}(\mathbf{x}_{t,N}^{cls}), \quad (3)$$

where $\mathbf{z}_t$ is the representation of frame t and $\mathbf{x}_{t,K}^{cls}$ is the CLS token of the last layer's sequence of the video encoder. To obtain the final video representation $\mathbf{v}$, the per-frame representation $\mathbf{v}_t$ are gathered to video level via Temporal Pooling:

$$\mathbf{v} = \mathbf{TemporalPooling}(\mathbf{v_t}). \quad (4)$$

Video Text Multi-modal Prompt Learning

Video Prompt Learning According to the prompt embedding process in deep language prompt tuning and VPT-Deep, we introduce b learnable prompt vectors $\widetilde{\mathbf{P}}$ =

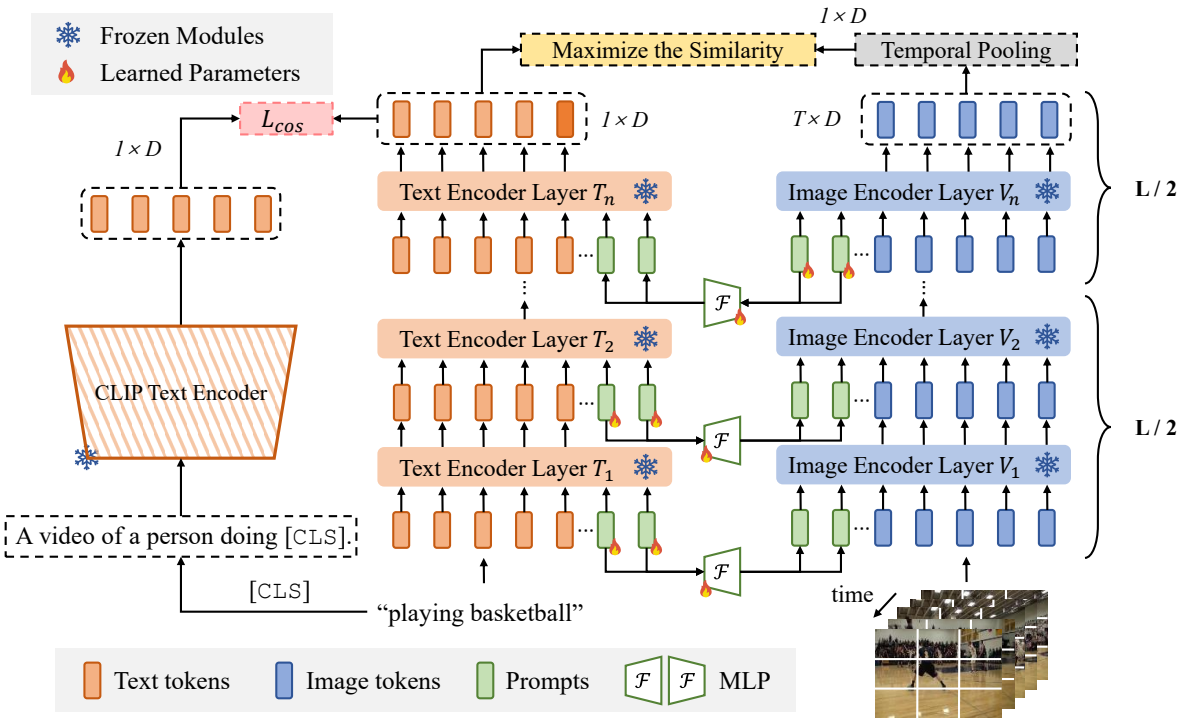

Figure 2: The architecture of the Video and Language Tuning CLIP (ViLT-CLIP) approach via Multimodal Prompt Learning. It is applied to the CLIP text encoder and image encoder. ViLT-CLIP tunes the vision and language branches, where only the input prompt tokens are learned while the rest of the model will be frozen.

$[\mathbf{P}^1, \mathbf{P}^2, \cdots, \mathbf{P}^b]$ in the vision branch of CLIP. New learnable prompt tokens are further introduced in the first K transformer layers of the image encoder $\mathcal{V}$.

$$\begin{aligned}[\mathbf{E}_{t,i}, \_\_] &= \boldsymbol{\mathcal{V}_i}([\mathbf{E}_{t,i-1}, \widetilde{\mathbf{P}}_{\boldsymbol{i-1}}]) \; i = 1, \cdots, K, \\ [\mathbf{E}_{t,j}, \widetilde{\mathbf{P}}_i] &= \boldsymbol{\mathcal{V}_j}([\mathbf{E}_{t,j-1}, \widetilde{\mathbf{P}}_{j-1}]) \; j = K+1, \cdots, N,\end{aligned} \tag{5}$$

Similar to **Video Encoder**, the final per-frame representation is the output CLS token $\mathbf{x}_{t,N}^{cls}$ of the last layer ($\mathcal{V}_N$), and projected to a V-L representation space using a linear projection $\texttt{ImageProj}$,

$$\mathbf{v}_t = \texttt{ImageProj}(\mathbf{x}_{t,N}^{cls}). \tag{6}$$

The final video-level representation is calculated by a **TemporalPooling** operation on the temporal dimension of image-level features following (4). These learnable prompt tokens are sufficient to catch the critical temporal information in the video sequence.

Text Prompt Learning Inspired by CoOp, we introduce b learnable tokens $\mathbf{P} = [\mathbf{P}^1, \mathbf{P}^2, \cdots, \mathbf{P}^b]$ in the text branch of CLIP to learn language context representation. The input embedding in the first layer now follows the form $[\mathbf{W}_0, \mathbf{P}]$, where $\mathbf{W}_0 = [\mathbf{w}_0^1, \mathbf{w}_0^2, \cdots, \mathbf{w}_0^M]$ corresponds to fixed input tokens embedded by the tokenizer of text branch. To learn language context prompts in each layer independently, new learnable vectors are introduced to each transformer block until a specific K^{th} layer ($\mathcal{T}_K$),

$$[\mathbf{W}_i, \_\_] = \boldsymbol{\mathcal{T}_i}([\mathbf{W}_{i-1}, \mathbf{P}_{\boldsymbol{i-1}}]) \quad i = 1, 2, \cdots, K, \tag{7}$$

Where $[\cdot, \cdot]$ refers to the concatenation operation on the sequence dim. To maintain the generalization ability of CLIP, we do not introduce new tokens after K^{th} transformer layers. Instead, the subsequent layer directly processes the previous layer's output, and final text representation $\mathbf{c}$ is obtained via a projection operation,

$$[\mathbf{W}_j, \mathbf{P}_j^t] = \boldsymbol{\mathcal{T}_j}([\mathbf{W}_{j-1}, \mathbf{P}_{j-1}]) \quad j = K+1, \cdots, N, \tag{8}$$

$$\mathbf{c} = \texttt{TextProj}(\mathbf{w}_N^M). \tag{9}$$

When $K = 1$, the learnable prompt tokens P_0 are only embedded in the input of the first transformer layer while other layers stay the same. This deep language prompt tuning approach degenerates to the shallow prompt technique in CoOp (Zhou et al. 2022b) and A5 (Ju et al. 2022).

Video Text Multimodal Prompt Learing CLIP is essentially cross-modal and maps V-L modalities to the same representation space. Benefiting from this, we propose a prompt learning approach called multi-modal prompt learning, which utilizes a single linear network $\mathcal{F}$ to associate language prompts with visual prompts to achieve unified representation learning. We believe providing learnable prompt vectors for language and vision modalities while holding a unified representation is necessary.

We discover that the text token in the shallow layer of the text branch contains essential textual information, which can guide the shallow vision transformer layer to learn spatial location information related to the category representation. It helps the visual transformer to catch the critical regions

according to language semantic context. Similarly, the deep layer tokens in the visual transformer have relatively affluent action representation. It is mapped as the textual prompt tokens embedding into the language branch, compensating for the poor category text description. As a result, similar category description is easier to be distinguished with additional captions from the visual branch. So we proposed bidirectional linear layers $\boldsymbol{\mathcal{F}}$ and $\widetilde{\boldsymbol{\mathcal{F}}}$ as follow:

$$\begin{aligned}
&[\mathbf{W}_i, __] = \boldsymbol{\mathcal{T}_i}([\mathbf{W}_{i-1}, \mathbf{P}_{i-1}]) \quad i = 1, \cdots, K, \\
&[\mathbf{W}_j, \mathbf{P}_j] = \boldsymbol{\mathcal{T}_j}([\mathbf{W}_{j-1}, \widetilde{\boldsymbol{\mathcal{F}}}(\widetilde{\mathbf{P}}_{j-1})]) \ \ j = K+1, \cdots, N.
\end{aligned} \tag{10}$$

$$\begin{aligned}
&[\mathbf{E}_{t,i}, __] = \boldsymbol{\mathcal{V}_i}([\mathbf{E}_{t,i-1}, \boldsymbol{\mathcal{F}}(\mathbf{P}_{i-1})) \quad i = 1, \cdots, K, \\
&[\mathbf{E}_{t,j}, P_j] = \boldsymbol{\mathcal{V}_j}([\mathbf{E}_{t,j-1}, \widetilde{\mathbf{P}}_{j-1}]) \quad j = K+1, \cdots, N.
\end{aligned} \tag{11}$$

Specifically, on the first K transformer layers, $\mathcal{F}$ maps the text prompting toekns $\mathbf{P}$ to the visual prompting tokens, where $\mathbf{P}$ and $\mathcal{F}$ are learnable. On the contrary, the visual prompt tokens $\widetilde{\mathbf{P}}$ in the later $N-K$ layers of the visual branch are transformed to the language prompting tokens through $\widetilde{\mathcal{F}}$, the learnable parameters are $\widetilde{\mathbf{P}}$ and $\widetilde{\mathcal{F}}$.

Video Scenario-Guided Optimization

The branch-aware hierarchical prompts learning affluent context in each transformer layer for better zero-shot and few-shot performance. However, plenty of prompts tend to lose the original category distribution and overfit to known categories. Inspired by (Yao, Zhang, and Xu 2023), we enhance the similarity between learnable prompts and hand-crafted prompts (`a video of a person doing <category>.`), which can alleviate forgetting of the scenario knowledge in the textual branch. The textual embedding generated by the CLIP and ViLT is defined to $\mathbf{c}_{clip} = \mathcal{T}(\mathbf{c}_i^{clip})$ and $\mathbf{c} = \mathcal{T}(\mathbf{c}_i)$, where $\mathbf{c}_{clip}$ represent the tokenized textual tokens in CLIP, and $\mathbf{c}_i = \{\mathbf{P}^1, \mathbf{P}^2, ..., \mathbf{P}^b, \mathbf{c}_i\}$ is the learnable prompt of the i-th class. We minimize the distance between $\mathbf{c}$ and $\mathbf{c}_{clip}$ as the ancillary loss for constraining the video context of the language prompting,

$$\mathcal{L}_{cos} = \frac{1}{N_c} \sum_{i=1}^{N} \frac{\mathbf{c} \cdot \mathbf{c}_{clip}}{\|\mathbf{c}\|_2 \cdot \|\mathbf{c}_{clip}\|_2}. \tag{12}$$

Training Loss

During training, we jointly optimize the text prompt tokens, visual prompt tokens, and the cooperative interaction module $\mathcal{F}$ between them, constraining the model to learn higher similarity between the extracted video clip features and the corresponding class embedding and lower similarity with other classes. The standard contrastive loss is:

$$\mathcal{L}_{ce} = -\sum_{\mathbf{v} \in \mathbf{V}} \log \frac{\exp(\cos(\mathbf{v}, \mathbf{t}_i,)/\tau)}{\sum_{i=1}^{N_c} \exp(\cos(\mathbf{v}, \mathbf{t}_i)/\tau)}, \tag{13}$$

where $\mathbf{v}$ is the video features, $\mathbf{t_i}$ is the embeddings of the class i and N_c is the number of seen classes. Combining with the similarity loss $\mathcal{L}_{cos}$, the final objective is:

$$\mathcal{L} = \mathcal{L}_{ce} + \alpha \mathcal{L}_{cos}, \tag{14}$$

where α is used to balance the effect of $\mathcal{L}_{cos}$ on the performance.

Experiment and Analysis

Experimental Setup and Protocols

Implementation Details In our experiments, all videos are decoded to 30 fps, and all randomly sampled frames are pre-processed to a spatial size of 224×224. In our experiments, we use handcrafted text prompts with a template "a video of a person doing `<category>`." The CLIP image (ViT-B/16) and the text encoder are frozen during training. Only the prompt tokens and the single linear network $\mathcal{F}$ are learnable parameters. Following CLIP (Radford et al. 2021), the maximum number of text tokens is set to 77, and the temperature hyperparameter τ is set to 0.07. We use an AdamW optimizer and weight decay of 0.001. We vary the epochs, batch size, and learning rate across different settings, with the specific values detailed below.

Datasets In the supervised setting, we train on the train set of Kinetics-400. For **few-shot experiments**, we create a K-shot split, where K-shot means randomly selected K samples from each category for training. Specifically, we use 2, 4, 8 and 16 shots for three datasets, HMBD-51, UCF-101 and SSv2. The training and validation processes consider only the first split in HMDB-51 and UCF-101, while for Kinetics and SSv2, the models are evaluated on their entire validation split. The setting uses 32 frames and evaluates with a single-view inference. For **base-to-novel generalization experiments**, we follow (Rasheed et al. 2023), using the most frequently occurring categories as the base classes and the rarely occurring categories as the novel classes, each containing randomly sampled 16-shots of every category. The model is evaluated on the first validation split of HMDB-51 and UCF-101 and the full validation split of SSv2. Base-to-novel and few-shot settings use 32 frames and are evaluated with a single-view inference.

Hyperparameters For action recognition in the few-shot setting, we prompt tuning ViLT-CLIP using 10 vision and language prompt tokens in all transformer layers and set half layer ($K = N/2$) of prompt tokens to constrain the learning of visual prompt tokens, and the visual prompt tokens in other half layers guide the learning of textual prompt tokens. We train the model for 30 epochs with a cosine decay scheduler and an initial learning rate of 8×10^{-3}. For action recognition in the base-to-novel generalization setting, we use 16 vision and language prompt tokens in the first 10 transformer layers of CLIP's vision and language encoders. We train for 12 epochs with a batch size of 64 and a learning rate of 4×10^{-2}. Similarly, we set $K = N/2$ to achieve balanced performance. All experiments can be done on four 11G NVIDIA TITAN Xp GPUs.

ViLT-CLIP Bridging Domain Gap!

We explore the ability of simple prompt tuning methods ViLT-CLIP with scenario guided to bridge domain gaps in a video action recognition task in different experimental settings: 1) fully supervised setting, 2) few-shot setting.

(i) Fully-Supervised Setting We compare the performance of ViLT-CLIP trained on Kinetics-400 with other unimodal and prompting methods in Table 2. We achieve **0.7%**

Model	HMDB-51				UCF-101				SSv2			
	K=2	K=4	K=8	K=16	K=2	K=4	K=8	K=16	K=2	K=4	K=8	K=16
Adapting pre-trained image VL models												
Vanilla CLIP (Radford et al. 2021)	41.9	41.9	41.9	41.9	63.6	63.6	63.6	63.6	2.7	2.7	2.7	2.7
ActionCLIP (Wang et al. 2021)	47.5	57.9	57.3	59.1	70.6	71.5	73.0	91.4	4.1	5.8	8.4	11.1
X-CLIP (Ni et al. 2022)	53.0	57.3	62.8	64.0	48.5	75.6	83.7	91.4	3.9	4.5	6.8	10.0
A5 (Ju et al. 2022)	39.7	50.7	56.0	62.4	71.4	79.9	85.7	89.9	4.4	5.1	6.1	9.7
Tuning pre-trained image VL models												
CLIP image-FT (Rasheed et al. 2023)	49.6	54.9	57.8	62.0	74.4	79.1	85.3	90.5	4.9	6.0	7.2	10.4
CLIP text-FT (Rasheed et al. 2023)	54.5	61.6	63.1	65.0	80.1	82.8	85.8	88.1	6.2	6.1	6.3	9.1
ViFi-CLIP (Rasheed et al. 2023)	57.2	**62.7**	64.5	66.8	80.7	85.1	90.0	92.7	6.2	7.4	8.5	12.4
Prompting pre-trained image VL models												
1: Independent V-L Prompting	57.8	60.0	64.8	67.4	84.0	87.1	90.4	93.1	7.3	8.1	9.8	13.0
2: Multimodal V-L Prompting	58.8	61.6	66.2	68.0	85.2	87.9	91.0	93.8	7.7	9.2	10.2	13.0
3: ViLT-CLIP	**60.6**	61.9	**66.9**	**69.6**	**85.3**	**90.0**	**91.3**	**93.8**	**7.8**	**9.4**	**10.3**	**13.2**
	+3.4	-0.8	+2.4	+2.8	+4.6	+4.9	+1.3	+0.8	+1.6	+2.0	+1.8	+0.8

Table 1: Few-shot setting: We compare ViLT-CLIP with approaches that adapt CLIP for video recognition on HMDB-51, UCF-101, and SSv2. In most shots, ViLT-CLIP achieves better performance against all the provided methods. Gains over the previous best are highlighted in +%. Some decreases are indicated in -%.

Method	Frames	Views	Top-1	Top-5	GFLOPs	TP
Uni-modal architectures						
Uniformer-B (2022b)	32	4 × 3	83.0	95.4	259	-
TimeSformer-L (2021)	96	1 × 3	80.7	94.7	2380	-
Swin-L (2022c)	32	4 × 3	83.1	95.9	604	-
Adapting pre-trained image VL models						
ActionCLIP (2021)	32	10 × 3	83.8	96.2	563	67.7
X-CLIP (2022)	16	4 × 3	**84.7**	**96.8**	287	58.5
A6 (2022)	16	-	76.9	93.5	-	-
Prompting pre-trained image VL models						
ViLT-CLIP B/16	16	4 × 3	77.6	94.5	287	72.1

Table 2: Fully-supervised setting: We compare ViLT-CLIP with uni-modal methods specifically designed for video action recognition and modals using extra components to adapt CLIP model to the video domain on K-400.

better top-1 accuracy against the A5 (Ju et al. 2022) prompting method using a frozen vision encoder. The simple approach with two-sided full multimodal prompt learning, and scenario-guided optimization gives ViLT-CLIP a more competitive performance compared to approaches that use additional learnable components for video-specific modeling.

(ii) Few-Shot Setting In Table 1, we compare ViLT-CLIP with 1) methods that adapt image-based multimodal VL model for video action recognition. 2) directly fine-tuning the text, visual, or both of multimodal VL model to video action recognition tasks. In addition, incorporating video-specific components to adapt CLIP can improve the generalization ability, highlighting the significance of minimizing the gap between different modalities. However, a simple prompt tuning method can better bridge the domain gap without compromising the generalization capabilities learned during the pre-training phase of CLIP. Here, we show the performance of the ViLT-CLIP in the few-shot setting. We observed that the effect of ViLT-CLIP gets better with increased shots. ViLT-CLIP outperforms all comparison methods for most shots (K = 2, 4, 6, 8) in the three datasets, HMDB-51, UCF-101, and SSv2. It achieves relatively significant gains with minimal data, suggesting it is robust to overfitting. Intuitively, comparing ViLT-CLIP with the previous best methods, we achieve **+4.6%** and **3.4%** gains in UCF-101 and HMDB-51.

Method	HMDB-51	UCF-101
Uni-modal zero-shot action recognition models		
ZSECOC (Qin et al. 2017)	22.6 ± 1.2	15.1 ± 1.7
UR (Zhu et al. 2018)	24.4 ± 1.6	17.5 ± 1.6
E2E (Brattoli et al. 2020)	32.7	48
ER-ZSAR (Chen and Huang 2021)	35.3 ± 4.6	51.8 ± 2.9
Adapting pre-trained image VL models		
Vanilla CLIP (Radford et al. 2021)	40.8 ± 0.3	63.2 ± 0.2
ActionCLIP (Wang et al. 2021)	40.8 ± 5.4	58.3 ± 3.4
X-CLIP (Ni et al. 2022)	44.6 ± 5.2	72.0 ± 2.3
A5 (Ju et al. 2022)	44.3 ± 2.2	69.3 ± 4.2
Prompting pre-trained image VL models		
ViLT-CLIP B/16	**45.3 ± 0.9**	**73.6 ± 1.1**

Table 3: Zero-Shot setting: We compare ViLT-CLIP with uni-modal methods designed specifically for zero-shot action recognition and approaches that adapt CLIP for video recognition on HMDB-51 and UCF-101.

Method	HMDB-51			UCF-101			SSv2			Kinetics-400		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
Adapting pre-trained image VL models												
Vanilla CLIP (Radford et al. 2021)	53.3	46.8	49.8	78.5	63.6	70.3	4.9	5.3	5.1	62.3	53.4	57.5
ActionCLIP (Wang et al. 2021)	69.1	37.3	48.5	90.1	58.1	70.7	13.3	10.1	11.5	61.0	46.2	52.6
X-CLIP (Ni et al. 2022)	69.4	45.5	55.0	89.9	58.9	71.2	8.5	6.6	7.4	74.1	56.4	64.0
A5 (Ju et al. 2022)	46.2	16.0	23.8	90.5	40.4	55.8	8.3	5.3	6.4	69.7	37.6	48.8
Tuning pre-trained image VL models												
CLIP image-FT (Rasheed et al. 2023)	62.6	47.5	54.0	86.4	65.3	74.4	9.2	8.5	8.8	72.9	58.0	64.6
CLIP text-FT (Rasheed et al. 2023)	70.0	51.2	59.1	90.9	67.4	78.3	12.4	9.5	10.8	73.4	59.7	65.8
ViFi-CLIP (Rasheed et al. 2023)	73.8	53.3	61.9	92.9	67.7	78.3	16.2	12.1	13.9	76.4	61.1	67.9
Prompt tuning pre-trained image VL models												
VL prompting (Rasheed et al. 2023)	77.1	54.9	64.1	95.1	74.1	83.6	15.8	11.5	13.3	–	–	–
1: Independent V-L Prompting	75.1	55.8	64.1	95.3	67.8	79.3	16.5	12.4	14.1	77.2	61.0	68.2
2: Multimodal V-L Prompting	75.8	57.2	65.2	**95.6**	68.4	80.2	17.3	12.6	14.6	**77.8**	61.4	68.6
3: ViLT-CLIP	**76.7**	**57.5**	**65.7**	95.2	**70.5**	**81.0**	**17.3**	**12.8**	**14.7**	77.4	**63.0**	**69.5**
	+2.9	+1.7	+3.8	+2.3	+3.2	+2.7	+1.1	+0.7	+0.8	+1.0	+1.9	+1.6

Table 4: Base-to-novel generalization: We compare the generalization ability of ViLT-CLIP with models that adapt CLIP for video tasks on HMDB-51, UCF-101, SSv2 and Kinetics-400. Here, HM refers to harmonic mean, which measures the trade-off between base and novel accuracy. ViLT-CLIP achieves strong generalization. Accuracy gains over prior best are shown in +%.

CLIP-ViLT Generalizes Better!

To analyze the generalization ability of the CLIP prompt-tuning approach, we evaluate two settings: 1) zero-shot setting to evaluate the cross-dataset generalization, and 2) base-to-novel generalization setting to test performance on novel categories. In the later setting, we use base-to-novel splits follow (Rasheed et al. 2023) for videos.

(i) Zero-Shot Setting For the zero-shot setting, we train the model on a large video action recognition dataset, Kinetics-400 and evaluate across HMDB-51, UCF101 on their corresponding three validation splits, and we report the top-1 average accuracy over them. It can be seen from Table 3 that we achieve **1.0%** and **1.6%** gains against the previous best method on HMDB-51 and UCF-101. We evaluated the methods on their corresponding three validation splits and calculated the average top-1 accuracy across them. In this setting, we evaluate with a single view of 32 frames.

(ii) Base-to-novel Generalization Setting In Table 4, we evaluate the generalization capability from base to novel categories on four datasets, HMDB-51, UCF-101, SSv2 and K-400. In comparison to the results of X-CLIP and ActionCLIP which use additional video-specific components to model temporal bias, ViLT-CLIP achieves higher base accuracy and shows remarkable improvements in novel accuracy. It offers a better balance between base and novel accuracy, resulting in the overall best harmonic mean across all datasets. For example, it achieves harmonic mean gains of **+3.8%** and **2.7%** in HMDB-51 and UCF-101 compared to prior best methods. Compared to VL prompting (Rasheed et al. 2023) method training without freezing any parameters, ViLT-CLIP gains **+0.8%** in SSv2. It shows that our scheme exhibits a competitive temporal understanding performance. The results indicate that multimodal prompting and scenario-guided optimization are effective in tuning the CLIP model to video tasks.

Method	R@1↑	R@5↑	MnR↓
CLIP Straight (Radford et al. 2021)	31.2	53.7	22.3
CLIP4Clip-meanP (Luo et al. 2022)	43.1	70.4	16.2
A5 (Ju et al. 2022)	36.7	64.6	–
ViLT-CLIP	**44.8**	**72.1**	**13.6**

Table 5: Comparison of ViLT-CLIP with methods explicitly prompting CLIP for videos on the Video-text retrieval task.

Video-Text Retrieval Tasks

Table 5 shows that ViLT-CLIP outperforms other methods on the MSR-VTT dataset. Noticed that our scheme is still competitive with existing methods focused on retrieval. Further details are given in the Appendix.

Conclusion

This work demonstrates the significance of a simple baseline for transferring the image-based CLIP model to the video domain. Existing solutions do not leverage the advantage of video-text joint prompt learning while not learning the knowledge from hand-craft templates and typically fine-tuning the CLIP backbone directly which lacks the balance between base-to-novel, few-shot generalization and fully supervised performance. Therefore, we propose a Video Language Tuning scheme with multimodal prompting and scenario-guided optimization to adapt the CLIP model for video tasks. We perform a comprehensive comparison and the results show that our scheme remains competitive in all settings while training a quite lower number of parameters.

Supplemental Material

We provide supplementary material that provides additional details and further qualitative analysis for the main paper. The contents follow the following order:

- Datasets Details
- Experiment and Analysis
- Additional Visualization

Dataset Details

Our analysis focuses on five established benchmarks for action recognition: Kinetics-400 (Carreira and Zisserman 2017), HMDB-51 (Kuehne et al. 2011), UCF-101 (Soomro, Zamir, and Shah 2012) and Something-Something v2 (SSv2) (Goyal et al. 2017).

Kinetics-400 and Kinetics-600: The Kinetics-400 datasets cover 400 human action categories and about 300,000 short videos. Each clip lasts about 10 seconds and is taken from different videos, of which 24,0436 and 19,796 videos serve as training and validation samples. These videos also cover finer-grained human activities and sports scenes from the YouTube website. Released in 2017, this dataset is currently the largest and most novel video recognition dataset.

HMDB-51: The HMDB-51 dataset consists of 6,766 short videos from 51 categories. These videos are collected from movies, the web, and public databases, covering various complex backgrounds and perspectives. There are three different training-validation splits, where 3,783 videos are used for training and 1,530 videos are used for validation.

UCF-101: UCF-101 contains 13,320 short video clips from 101 categories, collected from YouTube websites, containing various human activities and sports scenes. There are 3 different train-validation splits, all trained on 9,537 samples and validated on 3,783 samples.

Something-Something v2 (SSv2): The SSv2 dataset contains 220,847 video data in 174 classes, recording basic actions between humans and some everyday objects. The model needs to be trained, validated, and tested on 168,913, 24,777, and 27,157 clips, respectively. The dataset is mainly used to train models for the fine-grained understanding of human gestures, such as placing an object in a particular place or on another object. For evaluation metrics, we report the standard TOP1 accuracy.

We train on the 'Training-9K' split and test on the 'test 1k-A' split of the MSRVTT (Xu et al. 2016) dataset, consisting of 1,000 clip-text pairs. For LSMDC (Rohrbach et al. 2017), we train on 7,408 validation videos and evaluate another 1,000 videos. We follow the standard splitting in the MSVD (Chen and Dolan 2011) dataset, using 1,200 videos for training and 100 videos for validation. The Didemo (Hendricks et al. 2018) dataset consisted of 10,000 videos annotated with 40,000 sentences, and we evaluated video-text retrieval guided by (Bain et al. 2021).

MSRVTT: The MSR-VTT dataset is a large-scale dataset containing videos and corresponding subtitles released by Microsoft Research, consisting of 10,000 video clips from 20 categories, each annotated by 20 English sentences. The standard division is to use 7,000/9,000 video clips for training and 1,000 for testing.

MSVD: MSVD is published by Microsoft Research and contains 1,970 video clips, each containing 40 sentences. We used standard splitting, 1,200 videos for training, 100 videos for validation, and 670 videos for testing.

LSMDC: LSMDC covers 118,081 videos, each ranging from 2 to 30 seconds. The videos were extracted from 202 movies. The validation set contains 7,408 videos, and the test set has 1,000 videos from movie testing.

DiDeMo: The DiDeMo dataset contains 8,395, 1,065, and 1,004 videos for training and validation. The videos in the dataset were collected from Flickr, and each video was edited into a maximum of 30-second segments. The videos in the dataset are split into segments every 5 seconds to reduce the annotation complexity.

Experiment and Analysis

More Ablations

We add an ablation on the vision side prompting in Table 6. Here, We define a simple baseline, the zero-shot accuracy of naïve CLIP (Radford et al. 2021). Based on this, we add the text prompt P_t ($L = 10$) while keeping the rest of the model frozen. This achieves 72.97% top 1 accuracy on the K400. We then add the vision prompts P_v ($L = 10$), which improved the model's accuracy to 75.83%. After adding multimodal prompt linear, we achieve an accuracy of 76.46%. Finally, the scenario-guided optimizer brings the accuracy to 77.63%. This shows that the three prompting techniques are complementary to improving the performance of the model.

Text-Video Retrieval Setting

In Table 7, we show additional comparison results for retrieval benchmarks. Despite only optimizing a few parameters, our approach is comparable to the state-of-the-art methods in all metrics across the four datasets. For example, it achieves gains of +12.9%, 4.9% and 3.6% in MSVD, LSMDC, and Didemo compared to the prior prompting method. It reaches or exceeds the R@1 accuracy of CLIP4Clip, which is pre-trained on HowTo100M (136 million videos). For instance, it provides absolute benefits of +1.7% and 1.1% over CLIP4Clip on MSRVTT-9K and MSVD, which has a relatively small amount of data. This indicates that learning multimodal prompts is an effective

Method	Top-1
CLIP B/16 (Zero-shot)	40.10
ViLT-CLIP + P_t ($L = 10$)	72.97
ViLT-CLIP + P_t ($L = 10$) + P_v ($L = 10$)	75.83
ViLT-CLIP + P_t ($L = 10$) + P_v ($L = 10$) + MPL	76.46
ViLT-CLIP + P_t ($L = 10$) + P_v ($L = 10$) + MPL + Sg	77.63

Table 6: Ablations for different types of video prompts proposed in this work: Vision Prompts (P_v), Text Prompts (P_t) which is fixed to Unified Context for this ablation, Multimodal Prompt Learning (MPL) and Scenario-guided (Sg).

			MSRVTT (9K)			MSVD			LSMDC			DiDeMo		
Method	TrainD	E2E	R@1↑	R@5↑	MnR↓	R@1↑	R@5↑	MnR↓	R@1↑	R@5↑	MnR↓	R@1↑	R@5↑	MnR↓
CE (Liu et al. 2022b)	M	✗	21.7	51.8	28.2	19.8	49.0	–	12.1	27.8	94.3	16.1	41.1	43.7
MMT (Gabeur et al. 2020)	H+M	✗	26.6	57.1	24.0	–	–	–	12.9	29.9	75.0	–	–	–
TT-CE+ (Croitoru et al. 2021)	M	✗	29.6	61.6	–	25.4	56.9	–	17.2	36.5	–	21.6	48.6	–
Frozen (Bain et al. 2021)	CW+M	✓	31.0	59.5	–	33.7	64.7	–	15.0	30.8	–	34.6	65.0	–
Tuning pre-trained image VL models														
CLIP4Clip-meanP (Luo et al. 2022)	W+M	✓	43.1	70.4	16.2	46.2	76.1	10.0	20.7	38.9	65.3	43.4	70.2	17.5
Prompt tuning pre-trained image VL models														
CLIP-straight	W+M	✗	31.2	53.7	–	28.3	61.7	–	11.3	22.7	–	28.8	54.6	–
A5 (Ju et al. 2022)	W+M	✗	36.7	64.6	–	34.4	62.7	–	13.4	29.5	–	36.1	64.8	–
1: Independent V-L prompting	W+M	✗	44.3	69.6	18.4	46.4	76.3	9.8	17.6	36.7	69.1	38.8	65.8	16.7
2: Multimodal V-L prompting	W+M	✗	**44.8**	**72.1**	**13.6**	**47.3**	**76.8**	**9.8**	**18.3**	**37.1**	**66.5**	**39.7**	**66.5**	**15.3**

Table 7: Results of text-video retrieval. CLIP-straight refers to the original CLIP model with text query naïvely encoded, *i.e.* without using any prompt. E2E denotes if the model has been trained end-to-end. As these methods are pre-trained on different datasets with variable sizes, it is unlikely to make fair comparisons.

strategy for text-video retrieval in low-data scenarios. Since the text encoder from the pre-trained CLIP takes a limited number of textual tokens up to 77, whereas the text query of retrieval can be long, we only employ six learnable prompt vectors in both vision and language branches in these experiments. The results show that our scheme achieves remarkable performance on video text retrieval tasks using a much lower number of parameters.

Table 8 shows the results of our approach on the MSRVTT-9K dataset with different numbers of prompt tokens. As reported, we archive 8.1% better R@1 accuracy over the A5 prompting method, which learns language prompt tokens at the first transform layers and builds a transformer to make temporal modeling. Furthermore, We demonstrate the superiority of our baseline by embedding the [4+X] learnable prompt tokens at each layer of the text encoder. Thus, we report our main results using the [6+X] context textual prompts and visual prompts.

Method	T-P	V-P	R@1↑	R@5↑	MnR↓
CLIP (2021)	–	–	31.2	53.7	–
A5 (2022)	4 + X + 4	–	36.7	64.6	–
Independent V-L prompting	4 + X	X	39.1	61.7	19.0
	4 + X	4 + X	43.3	69.5	14.8
	4 + X	6 + X	42.6	69.7	14.6
	4 + X	8 + X	43.6	69.9	14.6
	6 + X	6 + X	43.5	69.6	14.2
	6 + X	10 + X	44.3	69.6	13.8
Multimodal V-L prompting	6 + X	6 + X	**44.8**	**71.4**	**13.6**

Table 8: Ablation study on the MSRVTT-9K dataset. Here, T-P represents the Textual Prompts, V-P refers to the Vision Prompts, and MnR stands for the Mean Rank.

Additional Visualization

Figure 3 shows t-SNE (Van der Maaten and Hinton 2008) visualizations of c_N video embeddings of [CLS] after the last Transformer layer for base-to-novel setting in HMDB-51 and UCF-101 datasets. All plots show that ViLT supports linearly differentiable representations. We also observe that the additional interactable way of each Transformer layer improves performance compared to visual prompt tuning with independent deep prompts.

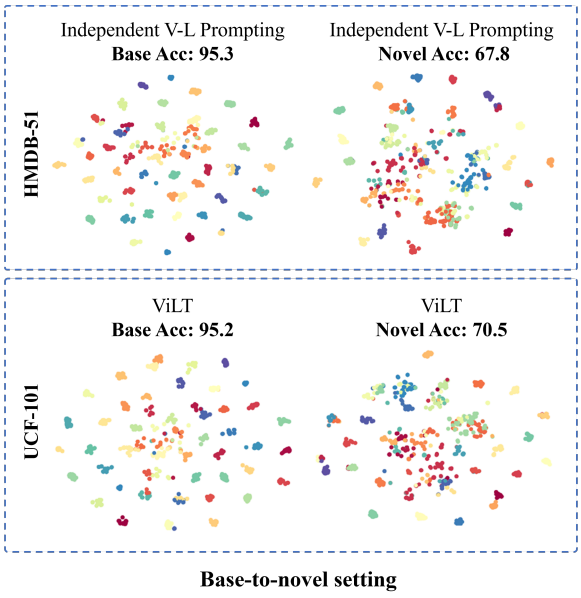

Figure 3: t-SNE (Van der Maaten and Hinton 2008) visualizations of the final image embedding c_N of our approach on the test set. We compare our method with independent prompting on classification separability visualizations on the HMDB-51 and UCF-101 datasets.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No.62076192), the Joint Fund Project of National Natural Science Foundation of China (No.U22B2054), in part by the China Postdoctoral Science Foundation (No.2023M742738), in part by the Program for Cheung Kong Scholars and Innovative Research Team in University (No.IRT_15R53), in part by The Fund for Foreign Scholars in University Research and Teaching Programs (the 111 Project) (No.B07048), in part by the Key Scientific Technological Innovation Research Project by Ministry of Education.

References

Bain, M.; Nagrani, A.; Varol, G.; and Zisserman, A. 2021. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 1728–1738.

Bertasius, G.; Wang, H.; and Torresani, L. 2021. Is space-time attention all you need for video understanding? In *ICML*, volume 2, 4.

Brattoli, B.; Tighe, J.; Zhdanov, F.; Perona, P.; and Chalupka, K. 2020. Rethinking zero-shot video classification: End-to-end training for realistic applications. In *CVPR*, 4613–4623.

Carreira, J.; and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 6299–6308.

Chen, D. L.; and Dolan, W. B. 2011. Collecting Highly Parallel Data for Paraphrase Evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL-2011)*. Portland, OR.

Chen, S.; and Huang, D. 2021. Elaborative rehearsal for zero-shot action recognition. In *ICCV*, 13638–13647.

Chen, S.; Zhao, Y.; Jin, Q.; and Wu, Q. 2020. Fine-grained video-text retrieval with hierarchical graph reasoning. In *CVPR*, 10638–10647.

Croitoru, I.; Bogolin, S.-V.; Leordeanu, M.; Jin, H.; Zisserman, A.; Albanie, S.; and Liu, Y. 2021. Teachtext: Cross-modal generalized distillation for text-video retrieval. In *ICCV*, 11583–11593.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*.

Fan, H.; Xiong, B.; Mangalam, K.; Li, Y.; Yan, Z.; Malik, J.; and Feichtenhofer, C. 2021. Multiscale Vision Transformers. In *ICCV*, 6824–6835.

Gabeur, V.; Sun, C.; Alahari, K.; and Schmid, C. 2020. Multi-modal transformer for video retrieval. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, 214–229. Springer.

Goyal, R.; Ebrahimi Kahou, S.; Michalski, V.; Materzynska, J.; Westphal, S.; Kim, H.; Haenel, V.; Fruend, I.; Yianilos, P.; Mueller-Freitag, M.; et al. 2017. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, 5842–5850.

Gu, X.; Lin, T.-Y.; Kuo, W.; and Cui, Y. 2021. Open-vocabulary object detection via vision and language knowledge distillation. In *ICLR*.

Hendricks, L. A.; Wang, O.; Shechtman, E.; Sivic, J.; Darrell, T.; and Russell, B. 2018. Localizing Moments in Video with Temporal Language. In *EMNLP*.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q.; Sung, Y.-H.; Li, Z.; and Duerig, T. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 4904–4916. PMLR.

Jia, M.; Tang, L.; Chen, B.-C.; Cardie, C.; Belongie, S.; Hariharan, B.; and Lim, S.-N. 2022. Visual prompt tuning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIII*, 709–727. Springer.

Ju, C.; Han, T.; Zheng, K.; Zhang, Y.; and Xie, W. 2022. Prompting visual-language models for efficient video understanding. In *ECCV*, 105–124. Springer.

Khattak, M. U.; Rasheed, H.; Maaz, M.; Khan, S.; and Khan, F. S. 2023. Maple: Multi-modal prompt learning. In *CVPR*, 19113–19122.

Kuehne, H.; Jhuang, H.; Garrote, E.; Poggio, T.; and Serre, T. 2011. HMDB: a large video database for human motion recognition. In *ICCV*, 2556–2563. IEEE.

Lei, J.; Li, L.; Zhou, L.; Gan, Z.; Berg, T. L.; Bansal, M.; and Liu, J. 2021. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, 7331–7341.

Li, B.; Weinberger, K. Q.; Belongie, S.; Koltun, V.; and Ranftl, R. 2022a. Language-driven Semantic Segmentation. In *ICLR*.

Li, K.; Wang, Y.; Gao, P.; Song, G.; Liu, Y.; Li, H.; and Qiao, Y. 2022b. niformer: Unified transformer for efficient spatiotemporal representation learning. In *ICLR*.

Li, P.; Liu, F.; Jiao, L.; Li, S.; Li, L.; Liu, X.; and Huang, X. 2023. Knowledge transduction for cross-domain few-shot learning. *Pattern Recognition*, 141: 109652.

Lin, J.; Gan, C.; and Han, S. 2019. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 7083–7093.

Liu, F.; Qian, X.; Jiao, L.; Zhang, X.; Li, L.; and Cui, Y. 2022a. Contrastive learning-based dual dynamic GCN for SAR image scene classification. *IEEE Transactions on Neural Networks and Learning Systems*.

Liu, Y.; Albanie, S.; Nagrani, A.; and Zisserman, A. 2022b. Use what you have: Video retrieval using representations from collaborative experts. In *ICLR*.

Liu, Z.; Ning, J.; Cao, Y.; Wei, Y.; Zhang, Z.; Lin, S.; and Hu, H. 2022c. Video Swin Transformer. In *CVPR*, 3202–3211.

Liu, Z.; Wang, L.; Wu, W.; Qian, C.; and Lu, T. 2021. Tam: Temporal adaptive module for video recognition. In *ICCV*, 13708–13718.

Luo, H.; Ji, L.; Zhong, M.; Chen, Y.; Lei, W.; Duan, N.; and Li, T. 2022. CLIP4Clip: An empirical study of CLIP for end to end video clip retrieval and captioning. *Neurocomputing*, 508: 293–304.

Ni, B.; Peng, H.; Chen, M.; Zhang, S.; Meng, G.; Fu, J.; Xiang, S.; and Ling, H. 2022. Expanding language-image pretrained models for general video recognition. In *ECCV*, 1–18. Springer.

Qin, J.; Liu, L.; Shao, L.; Shen, F.; Ni, B.; Chen, J.; and Wang, Y. 2017. Zero-shot action recognition with error-correcting output codes. In *CVPR*, 2833–2842.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Rao, Y.; Zhao, W.; Chen, G.; Tang, Y.; Zhu, Z.; Huang, G.; Zhou, J.; and Lu, J. 2022. Denseclip: Language-guided dense prediction with context-aware prompting. In *CVPR*, 18082–18091.

Rasheed, H.; Khattak, M. U.; Maaz, M.; Khan, S.; and Khan, F. S. 2023. Fine-tuned clip models are efficient video learners. In *CVPR*, 6545–6554.

Rohrbach, A.; Torabi, A.; Rohrbach, M.; Tandon, N.; Pal, C.; Larochelle, H.; Courville, A.; and Schiele, B. 2017. Movie description. *International Journal of Computer Vision*, 123: 94–120.

Simonyan, K.; and Zisserman, A. 2014. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27.

Soomro, K.; Zamir, A. R.; and Shah, M. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; and Paluri, M. 2015. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 4489–4497.

Tran, D.; Wang, H.; Torresani, L.; Ray, J.; LeCun, Y.; and Paluri, M. 2018. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 6450–6459.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; and Van Gool, L. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 20–36. Springer.

Wang, M.; Xing, J.; Liu, Y.; and Liu, Y. 2021. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*.

Wu, P.; He, X.; Tang, M.; Lv, Y.; and Liu, J. 2021. Hanet: Hierarchical alignment networks for video-text retrieval. In *Proceedings of the 29th ACM international conference on Multimedia*, 3518–3527.

Xu, H.; Ghosh, G.; Huang, P.-Y.; Okhonko, D.; Aghajanyan, A.; Metze, F.; Zettlemoyer, L.; and Feichtenhofer, C. 2021. Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *EMNLP*.

Xu, J.; Mei, T.; Yao, T.; and Rui, Y. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 5288–5296.

Yang, T.; Zhu, Y.; Xie, Y.; Zhang, A.; Chen, C.; and Li, M. 2023. AIM: Adapting Image Models for Efficient Video Action Recognition. In *ICLR*.

Yao, H.; Zhang, R.; and Xu, C. 2023. Visual-language prompt tuning with knowledge-guided context optimization. In *CVPR*, 6757–6767.

Yao, L.; Huang, R.; Hou, L.; Lu, G.; Niu, M.; Xu, H.; Liang, X.; Li, Z.; Jiang, X.; and Xu, C. 2021. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022a. Conditional prompt learning for vision-language models. In *CVPR*, 16816–16825.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9): 2337–2348.

Zhu, Y.; Long, Y.; Guan, Y.; Newsam, S.; and Shao, L. 2018. Towards universal representation for unseen action recognition. In *CVPR*, 9436–9445.