HiVLP: Hierarchical Interactive Video-Language Pre-Training

Bin Shao, Jianzhuang Liu, Renjing Pei,
Songcen Xu, Peng Dai, Juwei Lu, Weimian Li, Youliang Yan
Huawei Noah's Ark Lab
{shaobin3,liu.jianzhuang,peirenjing,xusongcen,
peng.dai,juwei.lu,liweimian,yanyouliang}@huawei.com

Abstract

Video-Language Pre-training (VLP) has become one of the most popular research topics in deep learning. However, compared to image-language pre-training, VLP has lagged far behind due to the lack of large amounts of video-text pairs. In this work, we train a VLP model with a hybrid of image-text and video-text pairs, which significantly outperforms pre-training with only the video-text pairs. Besides, existing methods usually model the cross-modal interaction using cross-attention between single-scale visual tokens and textual tokens. These visual features are either of low resolutions lacking fine-grained information, or of high resolutions without high-level semantics. To address the issue, we propose Hierarchical interactive Video-Language Pre-training (HiVLP) that efficiently uses a hierarchical visual feature group for multi-modal cross-attention during pre-training. In the hierarchical framework, low-resolution features are learned with focus on more global high-level semantic information, while high-resolution features carry fine-grained details. As a result, HiVLP has the ability to effectively learn both the global and fine-grained representations to achieve better alignment between video and text inputs. Furthermore, we design a hierarchical multi-scale vision contrastive loss for self-supervised learning to boost the interaction between them. Experimental results show that HiVLP establishes new state-of-the-art results in three downstream tasks, text-video retrieval, video-text retrieval, and video captioning.

1. Introduction

Recently, the framework of pre-training with large-scale uncurated data and then fine-tuning on some specific downstream tasks has attracted much attention. It firstly emerges in the field of Natural Language Processing (NLP), such as BERT [10], GPT [41] and T5 [42], which are pre-trained on a large corpus of web-scraped dataset and then fine-tuned on a wide variety of NLP downstream tasks.

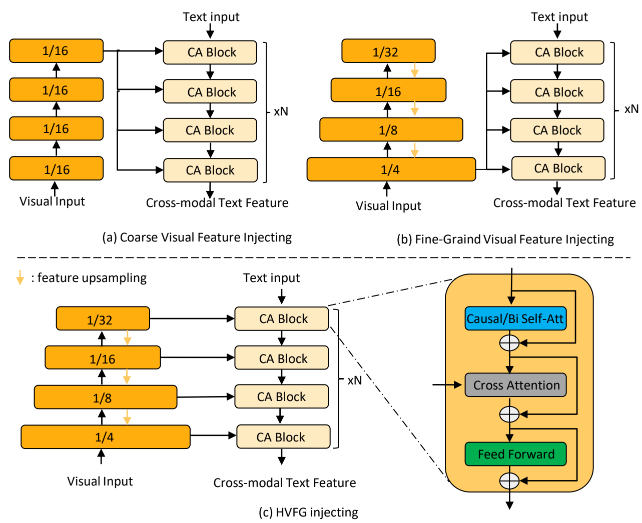

Figure 1. (a) Coarse visual feature injecting to CA blocks. (b) Fine-grained visual feature injecting to CA blocks. (c) Hierarchical visual feature injecting to CA blocks. A CA block consists of a casual/bi-directional self-attention layer, a cross-attention layer, and a feed forward layer. It is with bi-directional self-attention for vision-language understanding and with casual self-attention layer for vision-language generation.

Hereafter, it is transferred rapidly to the computer vision area. For examples, CLIP [40], ALIGN [16], Florence [58] and BLIP [20] all use more than 100 million open-domain image-text pairs in Image-Language Pre-training (ILP). However, most of the existing Video-Language Pre-training (VLP) works [47, 22, 51, 64, 30] use either a small-scale dataset (*e.g.*, YouCookII [63] with 14K video-text pairs) or a large-scale dataset with less diversity (*e.g.*, Howto100M [34] sourced from 1.22M videos). To solve this problem, we use a larger-scale dataset with 114M image-text pairs and a dataset with 2.5M video-text pairs to pre-train our model. We show that diversity is more important than the total amount of training pairs, and a small set of image-text pairs can achieve much better performance than using millions of video-text pairs. We believe this is a significant way to enhance VLP models and alleviate the cost of the collection of video-text pairs.

In VLP and ILP, existing works [47, 22, 64, 21, 56] often use cross-attention to model the cross-modal interaction between visual features and text features. However, they usually adopt only the single-scale and low-resolution visual features (*i.e.*, $\frac{1}{16}$ scale of the input) for cross-attention (CA) blocks, as shown in Figure 1(a). This scheme fails to obtain fine-grained interaction with text features and limits the performance of the pre-training model. For finer-grained interactions, [33] injects the high-resolution visual features (*i.e.*, $\frac{1}{4}$ scale of the input) to CA blocks as shown in Figure 1(b), but it does not have high-level semantics. To overcome these limitations, we propose Hierarchical interactive Video-Language Pre-training (HiVLP) that efficiently uses a Hierarchical Visual Feature Group (HVFG) for multi-modal cross-attention. As shown in Figure 1(c), HVFG includes different scales of visual features, where the low-resolution ones with high-level semantics are beneficial for global representation and the high-resolution ones with detailed information are useful for fine-grained interaction. Especially, HVFG is able to achieve much better accuracy because of such a multi-scale.

Many works [47, 64, 51] use self-supervised learning to assist the video-language pre-training by reconstructing the masked frame tokens. However, it may introduce noise to interactions between visual and textual features for the masked frame tokens [32]. In this paper, we propose a Multi-level Vision Contrastive (MVC) loss for our HiVLP by applying a global-to-local contrast learning to every scale in HVFG. The MVC loss does not damage the visual tokens and helps the multi-level alignment between visual and textual features.

Our contributions can be summarized as follows:

- To the best of our knowledge, our HiVLP is the first work that uses a hierarchical interaction for video-language pre-training. It is able to effectively learn both the global and fine-grained representations for better alignment between visual and textual features.
- We design a multi-level vision contrastive (MVC) loss for self-supervised learning that can sufficiently mine multi-level visual information to help video-language pre-training.
- We reveal that diversity is more important than the amount of training pairs, and using more diverse image-text pairs benefits a lot for VLP.
- Our HiVLP unifies video-language understanding and generation. It achieves state-of-the-art results in text-video retrieval, video-text retrieval, and video captioning.

2. Related Work

Image-Language Pre-Training (ILP). CLIP [40] is the pioneering work that collects large-scale web data (400M image-text pairs) and achieves competitive zero-shot performance on a variety of downstream tasks [39, 14]. ALIGN [16] is pre-trained with a larger-scale dataset (1.8B) obtaining better performance. FILIP [57] is pre-trained with 300M image-text pairs and designs a cross-modal late interaction mechanism for fine-grained contrastive learning. The key behind their success is that they take the advantages of large-scale datasets which are currently not available in VLP. To deal with this problem, we train our HiVLP model with a hybrid of video-text and image-text pairs.

Video-Language Pre-Training (VLP). Existing VLP works either use a pre-trained S3D [50] to extract visual features as vision input [30, 52, 51] to speed up the training process, or firstly perform pre-training on video-text pairs and then transfer the model to video-language generation tasks [47, 64, 22]. However, both these two training approaches limit the model performance because they are not trained end-to-end [24]. Our HiVLP jointly trains the model with image-text and video-text pairs end-to-end, and unifies both video-language understanding and generation in one framework. FiT [4] also involves pre-training with both image-text and video-text pairs, but can not do video-language generation.

Self-Supervised Learning (SSL). To effectively use datasets, many works [47, 64, 51, 49, 55] use self-supervised learning to assist VLP. VideoBERT [47] tokenizes video frames by hierarchical vector quantization, and then performs SSL by predicting the masked visual tokens. ActBERT [64] predicts the action and object words of masked video tokens. VLM [51] masks either all video tokens or all text tokens, and then uses tokens from one modality to recover masked tokens. However, the existing methods may damage the visual tokens, which introduces noise into cross-attention between visual and textual tokens [32]. For better visual representation learning and avoiding damaging visual tokens, we introduce the MVC loss which maximizes the mutual information between multi-scale global and local representations, and improves the multi-level alignment between visual and textual features.

Hierarchical Interaction. As far as we know, there is no related work about hierarchical interaction in VLP. The most related work is VinVL [59] in ILP. VinVL [59] uses an object detector to extract different sizes of objects as visual features to do cross-attention with textual features. The method of VinVL is complicated and the accuracy of its visual features is limited by the object detector. However, HiVLP makes different scales of visual features interact with textual features without the need of an object detector, which is more effective.

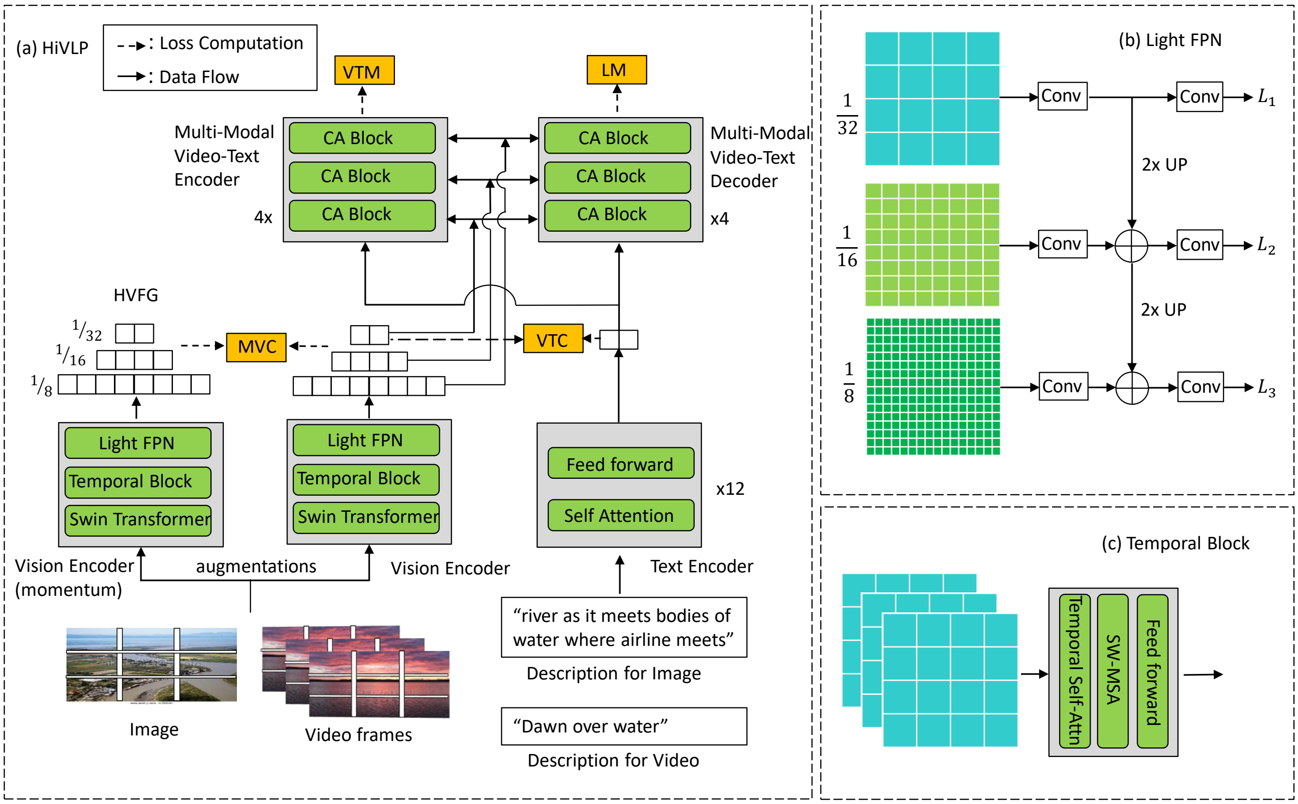

Figure 2. (a) Overview of HiVLP which consists of a vision encoder, a text encoder, a multi-modal video-text encoder and a multi-modal video-text decoder. Each image (or video clip) is transformed into two augmentations as the inputs to the vision encoder and the momentum vision encoder, respectively. The hierarchical visual feature group (HVFG) has scales $\{\frac{1}{8}, \frac{1}{16}, \frac{1}{32}\}$. The multi-level visual contrastive (MVC) loss is for multi-level SSL. Besides, HiVLP also uses the visual features of $\frac{1}{32}$ scale from the vision encoder and the output of the text encoder to optimize the vision-text contrastive (VTC) loss. Additionally, the multi-modal video-text encoder is trained with a vision-text matching (VTM) loss for the matching of video-/image-text pairs. The multi-modal video-text decoder is trained with a language modeling (LM) loss for video captioning. (b) The light FPN [25] is used for gradually upsampling and fusing the output features of the Swin Transformer. (c) The temporal block consists of a temporal self-attention layer, a SW-MSA layer [29] and a feed forward layer. We only replace the last block of the Swin Transformer with the temporal block for less parameters.

3. Approach

We firstly describe the detailed architecture of HiVLP. Then we present the hierarchical interaction HVFG. Finally, the pre-training objectives are given.

3.1. Architecture

As illustrated in Figure 2(a), HiVLP consists of a vision encoder, a momentum vision encoder, a text encoder, a multi-modal video-text encoder, and a multi-modal video-text decoder. The vision encoder and the text encoder extract image and text features respectively, and do not need to interact with each other for a fast approximate nearest neighbor search in inference [33]. The multi-modal video-text encoder re-ranks the top-k video-text pairs by an additional MLP head that predicts whether they are matched or not. The multi-modal video-text decoder is used to perform the video-language generation (*e.g.*, video captioning). The momentum vision encoder has the same architecture as the vision encoder, and the weights are updated by a momentum-based moving average strategy as in MoCo [15].

The main component of the vision encoder is a hierarchical backbone Swin Transformer [29], followed by a light feature pyramid network (FPN) [25]. Besides, we add a temporal self-attention layer in the last block of Swin Transformer to capture the temporal information, as shown in Figure 2(c). In Figure 2(b), the light FPN uses the output features of $\frac{1}{32}$, $\frac{1}{16}$, and $\frac{1}{8}$ scales from Swin Transformer as inputs, and then gradually upsamples those features for feature fusion. Note that to reduce computation overhead, we do not use $\frac{1}{4}$ scale of visual features in HVFG.

The text encoder's architecture is the same as BERT [10], which consists of 12 transformer layers. The multi-modal video-text encoder has 12 cross-attention (CA) blocks each with a bi-directional self-attention layer. The multi-modal video-text decoder consists of 12 CA blocks each with a casual self-attention layer.

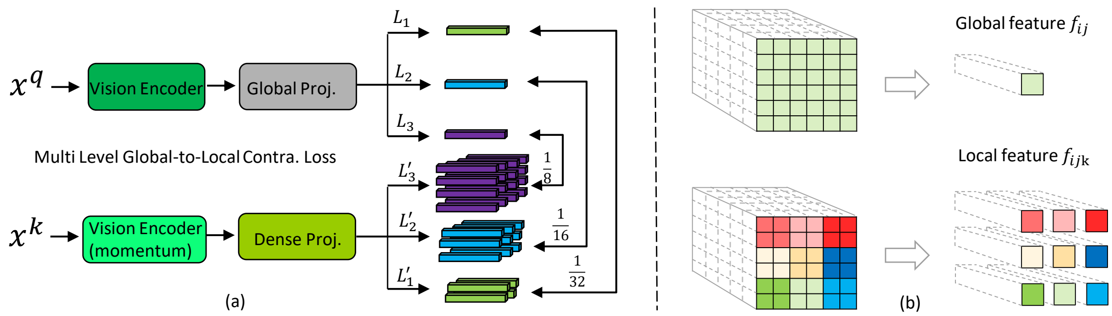

Figure 3. (a) Multi-level global-to-local contrastive loss. (b) Generation of global and local features.

3.2. Hierarchical Interaction

Existing works often inject single-scale visual features to CA blocks to interact with textual features. For example, ALBEF [21] and BLIP [20] use visual features of $\frac{1}{16}$ scale, which fail to do fine-grained cross-attention. In contrast, [33] adopts features of $\frac{1}{4}$ scale, but these over detailed features without high-level semantics. In our work, we propose a hierarchical interaction mechanism between visual and textual features via injecting HVFG into the CA blocks as shown in Figure 2(a). HVFG includes visual features of three scales $\{\frac{1}{8}, \frac{1}{16}, \frac{1}{32}\}$. The visual features of $\frac{1}{8}$ scale contain fine-grained information, and those of $\frac{1}{32}$ scale carry high-level semantics. Therefore, HVFG is able to obtain both global and fine-grained representations to boost the interaction between visual and textual features.

3.3. Pre-Training Objectives

The input of the vision encoder is an image or video clip $X \in R^{B\times M\times 3\times H\times W}$, which consists of M frames with the resolution $H \times W$, where M = 1 for images and B is the batch size. The input to the text encoder is a batch of tokenized sequences of words $T \in N^{B\times L}$, where L is the max length of a caption.

In our method, an input video clip X is split into non-overlapping patches $\{x_i\}_{i=1}^{BMN}, x_i \in R^{3\times P\times P}$, where $P \times P$ is input path size and $N = \frac{H\times W}{P\times P}$. The patches are tokenized by a linear embedding layer which is followed by a linear embedding layer of Swin Transformer.

Vision-Text Contrastive (VTC) Loss. VTC aims to pull positive pairs of vision and language representations together and push negative pairs far away. Like ALBEF [21] using a momentum distillation for vision-language pre-training, we introduce two queues to store the most recent K vision-text representations pairs from momentum encoders. Formally, the video-to-text contrastive loss is defined as:

$$L_{T2V} = \frac{1}{B}\sum_{i=1}^{B} log \frac{-exp(s(f_i, t_i^+)/\rho)}{exp(s(f_i, t_i^+)/\rho) + \sum_{j=1}^{K} exp(s(f_i, t_j^-)/\rho)}, \tag{1}$$

where ρ is the temperature, t_i^+ and t_j^- are the text features of the positive and negative text samples for the i-th vision input X_i respectively, and f_i is the visual feature of $\frac{1}{32}$ scale from X_i. $s(f_i, t_i^+)$ denotes the cosine similarity between f_i and t_i^+ that are matched, and $s(f_i, t_j^-)$ denotes the similarity between f_i and t_j^- that are not matched. Symmetrically, the text-to-video contrastive loss is:

$$L_{V2T} = \frac{1}{B}\sum_{i=1}^{B} log \frac{-exp(s(t_i, f_i^+)/\rho)}{exp(s(t_i, f_i^+)/\rho) + \sum_{j=1}^{K} exp(s(t_i, f_j^-)/\rho)}, \tag{2}$$

where f_i^+ and f_j^- are positive and negative visual features of the i-th text feature, respectively.

The sum of L_{V2T} and L_{T2V} serve as the vision-text contrastive loss:

$$L_{VTC} = \frac{1}{2}(L_{V2T} + L_{T2V}). \tag{3}$$

Multi-level Vision Contrastive (MVC) Loss. As shown in Figure 3(a), MVC aims to pull the global representation of a scale closer to those local patch representations from the same vision input in different views of a scale. The light FPN has three scales L_1, L_2, and L_3 (see Figure 2(b) and Figure 3(a)). The global feature f_{ij} of scale j of the i-th visual input is obtained by the average of all the patch features at L_j (Figure 3(b)). The local features $f_{ijk}, k = 1, 2, 3, ..., s_j$ are generated as shown in Figure 3(b) where f_{ijk} is the average feature pooling result of a local window such as 2×2, and s_j is the number of the local averaged features at $L_j^{'}$. Finally, the MVC loss is formulated as:

$$L_{MVC}^{ij} = \frac{exp(s(f_{ij}, \hat{f}_{ij}^+)/\rho)}{exp(s(f_{ij}, \hat{f}_{ij}^+)/\rho) + \sum_{k=1}^{(B-1)S_j} exp(s(f_{ij}, \hat{f}_{ijk}^-)/\rho)}, \tag{4}$$

$$L_{MVC} = -\frac{1}{3B}\sum_{i=1}^{B}\sum_{j=1}^{3} -logL_{MVC}^{ij}, \tag{5}$$

where $\hat{f}_{ij}$ and $\hat{f}_{ijk}$ correspond to f_{ij} and f_{ijk}, respectively, but from the other augmentation of the visual input.

Vision-Text Matching (VTM) Loss. Through a MLP layer, VTM is used to judge whether a video-/image-text pair is matched or not with the cross entropy function. There are B positive pairs from the batch, and $2B$ negative pairs are obtained according to the hard negative mining strategy in [21]. This loss is defined as:

$$L_{VTM} = -\frac{1}{3B}\sum_{i=1}^{3B}\left[y_i log(p_i) + (1-y_i) log(1-p_i)\right], \tag{6}$$

where y_i is the ground-truth, and p_i is the output of the MLP layer of the i-th pair.

Language Modeling (LM) Loss. We inject the hierarchical visual representations into the multi-modal video-text decoder to generate captions. Let i-th input text be $T_i = [t_i^1, t_i^2, ..., t_i^L]$. This loss is defined as:

$$L_{LM} = -\frac{1}{BL}\sum_{i=1}^{B}\sum_{l=1}^{L} log(p(t_i^l|t_i^{l-1}, ..., t_i^1, f_{i1}, f_{i2}, f_{i3})), \tag{7}$$

where $p(t_i^l|t_i^{l-1}, ..., t_i^1, f_{i1}, f_{i2}, f_{i3})$ is the output probability of the multi-modal video-text decoder for the l-th token t_i^l given the previously fed tokens $t_i^{l-1}, ..., t_i^1$ and the multi-level visual features f_{i1}, f_{i2}, f_{i3}.

Total Loss. Finally, we have the total loss to train HiVLP:

$$L_{total} = \alpha L_{VTC} + \beta L_{MVC} + \gamma L_{VTM} + \delta L_{LM}, \tag{8}$$

where α, β, γ, and δ are the weight coefficients of the losses.

4. Experiments

In this section, we first describe the pre-training and downstream datasets, and the implementation details. Then we show the experimental and comparison results, and finally ablate our model.

4.1. Datasets

Pre-Training Datasets. Following FiT [4], we involve two popular datasets (CC-3M [46] and WebVid-2M [4]) for pre-training HiVLP. CC-3M consists of 3M image-text pairs and WebVid-2M includes 2.5M video-text pairs, in total 5.5M image-/video-text pairs. Besides, we use LAION100M, D-14M, and WebVid-2M (total 116.5M image-/video-text pairs) as the larger-scale dataset to train our model, resulting in HiVLP*. LAION100M is a subset of LAION [45]. D-14M is a combined dataset of image-text pairs from COCO [26], Visual Genome [17], CC3M [46], CC12M [6], and SBU captions [35].

Downstream Task Datasets. There are four downstream datasets used in our work for video retrieval. (i) MSR-VTT [53] consists of 10K YouTube videos with 200K human-annotated descriptions. Similar to previous methods [4], we use the split of 9K for training and 1K for testing. (ii) MSVD [7] is a smaller dataset with 1970 videos and 78800 sentences, about 40 descriptions per video. Like [4], we use the split of 1200, 100, and 670 videos for training, validation and testing, respectively. (iii) DiDeMo [3] has 10k videos, each of which is described by multiple sentences, resulting in 40K sentences. For a fair comparison, we follow the setting in [4], where all descriptions for one video are concatenated into a single sentence. (iv) LSMDC [43] contains 128K clips. Like [4], we use 7408 clips for validation and 1000 clips for testing. For video captioning, we evaluate our model on MSR-VTT and MSVD. Following the split of [30], on MSR-VTT, we use $6.5K$ training videos and $2.9K$ testing videos; on MSVD, we use $1.2K$ training videos and 670 testing videos.

4.2. Implementation Details

Our model HiVLP is pre-trained on 16 GPUs (32G memory). Our vision encoder is initialized by Swin-B [29] pre-trained on ImageNet-21k [9]. The light FPN is randomly initialized. The text encoder, multi-modal video-text encoder, and multi-modal video-text decoder are BERTs initialized from BERTbase [10]. HiVLP is pre-trained for 25 epochs, the first 20 epochs only with image-text pairs with a batch size of 30 and the last 5 epochs with both image-text and video-text pairs. Each video clip consists of 4 frames. We use AdamW optimizer with a weight decay of 0.02, and the learning rate is initialized as 10^{-5} and is warmed up to 10^{-4} after 3,000 training iterations. We then decrease the learning rate by the cosine decay strategy to 10^{-5}. For the hyper-parameters in Equation 8, we set α, β, and δ to 1.0 and γ to 0.001. The input resolution is 224×224 with data augmentations used in [21]. And we use 160 GPUs (16G memory) to train HiVLP* with a batch size of 18 per GPU. For speeding up the training of HiVLP*, we sample 1 frame from each clip.

For the downstream task of text-video retrieval, we sample 4 frames per video for training. Since the multi-modal video-text encoder filters top-k candidates during inference (k is set to 128). We sample 8 frames per video for testing on MSR-VTT, MSVD, and LSMDC. Because the videos in DiDeMo are longer, we sample 10 frames for testing from it. For the downstream task of video captioning, we evaluate our model on MSR-VTT and MSVD, with 8 random frames per video for training and 16 frames per video for testing. For all downstream tasks, the initial learning rate is set to 5×10^{-6}, and the weight decay, the batch size, and the total number of epochs are set to 0.05, 64, and 10, respectively.

Method	Pre-Training Datasets	#Pairs	R@1↑	R@5↑	R@10↑	MedR↓	MR↑
Zero-Shot							
SupportSet [38]	HowTo100M	100M	8.7	23.0	31.1	31.0	20.9
HD-VILA [54]	HD-VILA-100M	100M	14.4	31.6	41.6	17.5	29.2
FiT [4]	CC3M, WebVid-2M	5.5M	18.7	39.5	51.6	10.0	36.6
ALPRO [19]	CC3M, WebVid-2M	5.5M	24.1	44.7	55.4	8.0	41.4
CLIP [40]	WIT400M	400M	30.6	54.4	64.3	4.0	49.8
Florence [58]	FLD-900M	900M	37.6	63.8	72.6	-	58
BLIP [20]	L-115M, D-14M	239M	43.3	65.6	74.7	**2.0**	61.2
HiVLP	CC3M, WebVid-2M	5.5M	26.4	47.3	55.7	7.0	43.1
HiVLP*	L-100M, D-14M, WebVid-2M	116.5M	**43.5**	**66.4**	**76.4**	**2.0**	**62.1**
Fine-Tuning							
ActBERT [64]	HowTo100M	100M	16.3	42.8	56.9	10.0	38.7
HERO [22]	HowTo100M	100M	16.8	43.4	57.7	-	39.3
NoiseEstimation [2]	HowTo100M	100M	17.4	41.6	53.6	8.0	37.5
UniVL [30]	HowTo100M	100M	21.2	49.6	63.1	6.0	44.6
ClipBERT [18]	HowTo100M	100M	22.0	46.8	59.9	6.0	42.9
AVLnet [44]	HowTo100M	100M	27.1	55.6	66.6	4.0	49.8
MMT [12]	HowTo100M	100M	26.6	57.1	69.6	4.0	51.1
SupportSet [38]	HowTo100M	100M	30.1	58.5	69.3	3.0	52.6
FiT [4]	CC3M, WebVid-2M	5.5M	31.0	59.5	70.5	3.0	53.7
ALPRO [19]	CC3M, WebVid-2M	5.5M	33.9	60.7	73.2	3.0	55.9
HD-VILA [54]	HD-VILA-100M	100M	35.0	65.2	77.2	3.0	59.1
CLIP4Clip [31]	WIT400M	400M	44.5	71.4	81.6	2.0	65.8
HiVLP	CC3M, WebVid-2M	5.5M	41.1	65.9	75.9	2.0	61.0
HiVLP*	L-100M, D-14M, WebVid-2M	116.5M	**50.9**	**76.4**	**83.6**	**1.0**	**70.3**

Table 1. Comparisons with SOTA text-to-video retrieval methods on MSR-VTT. R@K: Recall@K; MedR: Median Rank; MR: Mean Rank; L-115M: LAION115M; L-100M: LAION100M.

Method	R@1↑	R@5↑	R@10↑	MR↑
CE [28]	19.8	49.0	63.8	44.2
Support Set [38]	28.4	60.0	72.9	53.8
FiT [4]	33.7	64.7	76.3	57.3
CLIP4Clip [31]	46.2	76.1	84.6	68.9
HiVLP	39.1	68.1	77.2	61.5
HiVLP*	**50.2**	**78.9**	**85.8**	**71.6**

Table 2. Text-to-video results on the MSVD dataset.

Method	R@1↑	R@5↑	R@10↑	MR↑
MMT [12]	12.9	29.2	38.8	27.0
FiT [4]	15.0	30.8	39.8	28.5
MDMMT [11]	18.8	38.5	47.9	35.1
CLIP4Clip [31]	21.6	41.8	49.8	37.7
HiVLP	20.4	38.8	48.4	35.9
HiVLP*	**24.8**	**44.1**	**54.6**	**41.2**

Table 4. Text-to-video results on the LSMDC dataset.

Method	R@1↑	R@5↑	R@10↑	MR↑
ClipBERT [18]	20.4	44.5	56.7	40.5
FiT [4]	34.6	65.0	74.7	58.1
ALPRO [54]	35.9	67.5	78.8	60.7
CLIP4Clip [31]	43.4	70.2	80.6	64.7
HiVLP	40.5	68.7	77.1	62.1
HiVLP*	**48.1**	**73.8**	**82.5**	**68.1**

Table 3. Text-to-video results on the DiDeMo dataset.

4.3. Comparisons with State-of-the-Art Methods

In this section, we compare HiVLP with state-of-the-art (SOTA) methods on three popular video-language downstream tasks (text-video retrieval, video-text retrieval, and video captioning).

Text-Video Retrieval. We use MSR-VTT, MSVD, LSMDC, and DiDeMo datasets to evaluate text-video retrieval. We report zero-shot and fine-tuning results (Table 1) on MSR-VTT. For zero-shot retrieval, pre-trained with the same small amount of data, HiVLP outperforms FiT by a large margin (7.4% on R@1). CLIP, Florence, BLIP, and HiVLP* use large-scale datasets for pre-training, with the numbers of pairs 400M, 900M, 239M, and 116.5M, respectively. BLIP performs best among previous works. However, even with much fewer training pairs (116.5M vs. 239M), our HiVLP* outperforms BLIP on all the metrics except obtaining the same MedR. Note that BLIP uses both filtered and synthetic captions in LAION115M and D-14M, while we only use the filtered captions in LAION100M and D-14M.

For fine-tuning comparison, HiVLP/HiVLP* is fine-tuned with the VTC and VTM losses. In Table 1, we see that

Method	MSR-VTT			MSVD			LDMDC		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP4Clip-meanP$^+$ [31]	43.1	70.5	81.2	56.6	79.7	84.3	20.6	39.4	47.5
CLIP4Clip-seqTransf$^+$ [31]	42.7	70.9	80.6	62.0	87.3	92.6	20.8	39.0	48.6
X-Pool$^+$ [13]	44.4	73.3	84.0	66.4	**90.0**	94.2	22.7	42.6	51.2
HiVLP*	**51.5**	**75.9**	**83.1**	**67.2**	89.5	**95.7**	**25.6**	**45.6**	**53.8**

Table 5. Comparison with SOTA methods on MSR-VTT, MSVD, and LSMDC for video-text retrieval. $^+$ indicates using both CLIP's image and text encoders as its encoders.

Method	#Test Frames	MSVD				MSR-VTT			
		B4↑	M↑	R↑	C↑	B4↑	M↑	R↑	C↑
SibNet [27]	30	54.2	34.8	71.7	88.2	40.9	27.5	60.2	47.5
SAAT [62]	28	46.5	33.5	69.4	81.0	39.9	27.7	61.2	51.0
STG-KD [36]	16	52.2	36.9	73.9	93.0	40.5	28.3	60.9	47.1
PMI-CAP [8]	32	54.6	36.4	-	95.1	42.1	28.7	-	49.4
ORG-TRL [61]	28	54.3	36.4	73.9	95.2	43.6	28.8	62.1	50.9
OpenBook [60]	28	-	-	-	-	33.9	23.7	50.2	52.9
SwinBERT [24]	64	66.3	42.4	80.9	149.4	45.4	30.6	64.1	55.9
HiVLP	16	67.1	**45.3**	81.8	144.5	47.4	31.2	64.6	62.3
HiVLP*	16	**68.3**	45.1	**82.0**	**151.6**	**49.2**	**32.4**	**65.9**	**67.8**

Table 6. Comparison with SOTA methods on MSVD and MSR-VTT for video captioning.

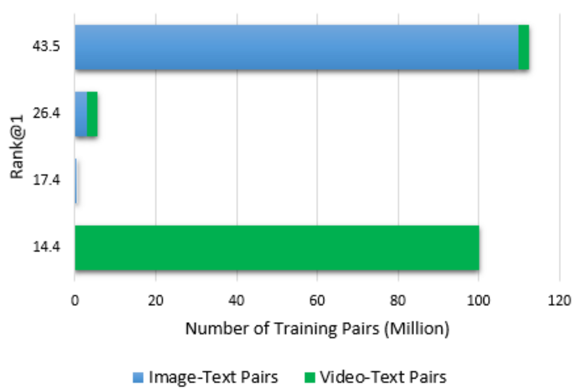

Figure 4. Diversity vs. amount of training pairs.

HiVLP uses the same pre-training datasets as FiT but again outperforms it by a large margin (10.1% on Rank@1). With only about 29% amount of pre-training pairs in CLIP4Clip, HiVLP* exceeds CLIP4Clip on all the metrics significantly. Tables 2, 3, and 4 show the comparison on another three datasets, where HiVLP* outperforms the other methods by a large margin.

Video-Text Retrieval. In Table 5, we use HiVLP* to compare with the SOTA methods which are built on CLIP. Our method can beat these CLIP-based methods with fewer pre-training pairs (116.5M vs. 400M).

Video Captioning. We also evaluate our model on MSR-VTT and MSVD for video captioning. Four popular metrics BLEU4 [37] (B4), METEOR [5] (M), ROUGE-L [23] (R), and CIDEr [48] (C) are employed. We fine-tune our model with the LM loss for the video-language generation task.

As shown in Table 6, HiVLP works best on all the metrics except one. Note that SwinBERT adopts 64 frames for testing.

4.4. Ablation Study

We randomly choose 1/10 image-text pairs from CC3M, resulting in 300k image-text pairs to do ablation study. We surprisingly find that our model pre-trained with only 300k image-text pairs can beat the models SupportSet and HD-VILA pre-trained respectively on Howto100M and HD-VILA-100M, when they are transferred to the MSR-VTT dataset for text-to-video zero-shot retrieval (45.3 vs. 31.1 vs. 41.6 on R@10).

Image-Text Pairs to Enhance VLP. As shown in Figure 4, when the number of image-text pairs is increased from 3M to 114M but with the same set of video-text pairs, the Rank@1 on the MSR-VTT dataset raises a lot (from 26.4 to 43.5); when only using much fewer image-text pairs of CC3M than video-text pairs of HD-VILP (0.3M vs. 100M), the Rank@1 performance is much better (17.4 vs. 14.4). It reveals that image-text pairs can bring more diversity than the same amount or more video-text pairs, and more image-text pairs benefit a lot. We believe this is a significant way to enhance the performance of VLP models.

MVC Loss Design. We use the HiVLP without the MVC loss as the baseline model. As shown in Table 7, the visual SSL both G2G and G2L-1 are able to improve the performance of the baseline. Using the G2L-1 contrastive loss is better than the G2G contrastive loss, because G2L-1 can dig out local information while G2G cannot. And using more number of different scales of visual features in the MVC loss (*i.e.*, G2L-2 and G2L-3) can further boost the align-

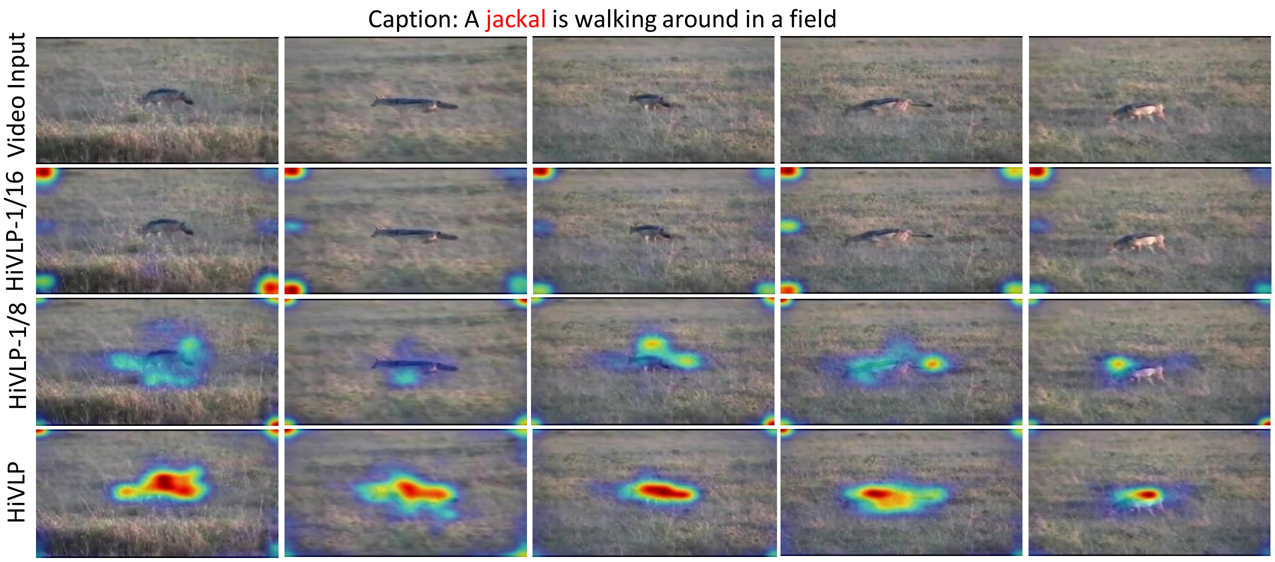

Figure 5. Grad-CAM visualization on the cross-attention maps of the different-scale visual features injected to the CA blocks. We extract the attention maps in the 8th layer of the multi-modal video-text encoder. The word "jackal" in the caption is the attention target in the video. (Best viewed on screen.)

Model	MVC loss	R@10↑	MR↑
Baseline	-	43.5	31.1
+G2G	$(L_1, L_1^{''})$	44.2	31.4
+G2L-1	$(L_1, L_1^{'})$	44.9	31.8
+G2L-2	$(L_1, L_1^{'}), (L_2, L_2^{'})$	45.2	32.2
+G2L-3	$(L_1, L_1^{'}), (L_2, L_2^{'}), (L_3, L_3^{'})$	**45.3**	**32.5**

Table 7. Impact of the MVC loss for zero-shot transfer to MSR-VTT. Note that $(L_j, L_j^{'})$ is the j scale of global-to-local (G2L) contrastive loss, with L_j, $L_j^{'}$, $j = 1, 2, 3$ indicated in Figure 3(a). $L_1^{''}$ denotes replacing the dense projection $L_1^{'}$ by the global projection as indicated in the upper part of Figure 3(b). And $(L_1, L_1^{''})$ denotes the $j = 1$ global-to-global (G2G) contrastive loss. MR: Mean Rank.

ment between the multi-scale visual features and the textual features.

Impact of HVFG. Table 8 compares our hierarchical vision features with only single-scale ones. HiVLP-$1/8$ uses large single scales of visual features to achieve the best performance among HiVLP-$1/32$, HiVLP-$1/16$, and HiVLP-$1/8$. However, HiVLP-$1/8$ does not efficiently use the deeper and small scales of visual features with higher-level semantics. To solve this problem, HiVLP effectively uses both global and fine-grained visual features to interact with textual features, resulting in much better performance than HiVLP-$1/8$ (32.5 vs. 30.5 on MR).

Visualization. As Figure 5 shown, HiVLP can capture fine-grained information better than HiVLP-$1/8$. The attention maps of HiVLP-$1/16$ look random due to its low-resolution feature maps.

Model	scales of features	R@1↑	R@10↑	MR↑
HiVLP-$1/32$	$\{\frac{1}{32}, \frac{1}{32}, \frac{1}{32}\}$	14.0	43.4	30.0
HiVLP-$1/16$	$\{\frac{1}{16}, \frac{1}{16}, \frac{1}{16}\}$	14.3	43.6	30.2
HiVLP-$1/8$	$\{\frac{1}{8}, \frac{1}{8}, \frac{1}{8}\}$	14.8	43.7	30.5
HiVLP	$\{\frac{1}{32}, \frac{1}{16}, \frac{1}{8}\}$	**17.4**	**45.3**	**32.5**

Table 8. Ablation study of different vision features injecting to CA blocks for text-to-video retrieval zero-shot transfer to MSR-VTT.

5. Conclusion

We have presented HiVLP, a novel hierarchical interactive video-language pre-training framework. Different from previous methods that input single-scale visual features to cross-attention blocks, HiVLP injects a hierarchical vision feature group (HVFG) to effectively use both global and fine-grained visual features for interaction with textual features. Additionally, our HiVLP is pre-trained with multi-level self-supervised learning that can further improve the model performance. We also reveal that VLP models benefit a lot from the diversity of image-text pairs. Extensive experimental results of downstream tasks (text-video retrieval, video-text retrieval, and video captioning) on 4 popular benchmark datasets show that HiVLP is able to achieve better performance than previous SOTA methods overall by a large margin.

Acknowledgements

We gratefully acknowledge the support of MindSpore [1], CANN (Compute Architecture for Neural Networks) and Ascend AI Processor used for this research.

References

[1] https://www.mindspore.cn/.

[2] Elad Amrani, Rami Ben-Ari, Daniel Rotman, and Alex Bronstein. Noise estimation using density estimation for self-supervised multimodal learning. In *AAAI*, 2021.

[3] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017.

[4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021.

[5] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL*, 2005.

[6] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021.

[7] David Chen and William Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011.

[8] Shaoxiang Chen, Wenhao Jiang, Wei Liu, and Yu-Gang Jiang. Learning modality interaction for temporal sentence localization and event captioning in videos. In *ECCV*, 2020.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

[11] Maksim Dzabraev, Maksim Kalashnikov, Stepan Komkov, and Aleksandr Petiushko. Mdmmt: Multidomain multimodal transformer for video retrieval. In *CVPR*, June 2021.

[12] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020.

[13] Satya Krishna Gorti, Noel Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. X-pool: Cross-modal language-video attention for text-video retrieval. In *CVPR*, 2022.

[14] Peiyan Guan, Renjing Pei, Bin Shao, Jianzhuang Liu, Weimian Li, Jiaxi Gu, Songcen Xu, Youliang Yan, and Edmund Lam. Pidro: Parallel isomeric attention with dynamic routing for text-video retrieval. In *ICCV*, 2023.

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[16] 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.

[17] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123(1):32–73, 2017.

[18] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, 2021.

[19] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven C.H. Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *CVPR*, 2022.

[20] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

[21] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*, 2021.

[22] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *EMNLP*, 2020.

[23] Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *ACL*, 2004.

[24] Kevin Lin, Linjie Li, Chung-Ching Lin, Faisal Ahmed, Zhe Gan, Zicheng Liu, Yumao Lu, and Lijuan Wang. Swinbert: End-to-end transformers with sparse attention for video captioning. In *CVPR*, 2022.

[25] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[27] Sheng Liu, Zhou Ren, and Junsong Yuan. Sibnet: Sibling convolutional encoder for video captioning. In *NeurIPS*, 2020.

[28] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. In *BMVC*, 2019.

[29] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.

[30] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv:2002.06353*, 2020.

[31] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv:2104.08860*, 2021.

[32] Jianjie Luo, Yehao Li, Yingwei Pan, Ting Yao, Hongyang Chao, and Tao Mei. Coco-bert: Improving video-language pre-training with contrastive cross-modal matching and denoising. In *ACM MM*, 2021.

[33] Antoine Miech, Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Andrew Zisserman. Thinking fast and slow: Ef-

ficient text-to-visual retrieval with transformers. In *CVPR*, 2021.

[34] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In *ICCV*, 2019.

[35] Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NeurIPS*, 2011.

[36] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. In *CVPR*, 2020.

[37] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002.

[38] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander Hauptmann, Joao Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. In *ICLR*, 2021.

[39] Renjing Pei, Jianzhuang Liu, Weimian Li, Bin Shao, Songcen Xu, Peng Dai, Juwei Lu, and Youliang Yan. Clipping: Distilling clip-based models with a student base for video-language retrieval. In *CVPR*, 2023.

[40] 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. *ICML*, 2021.

[41] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. In *ICML*, 2018.

[42] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020.

[43] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *CVPR*, 2015.

[44] Andrew Rouditchenko, Angie Boggust, David Harwath, Brian Chen, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Hilde Kuehne, Rameswar Panda, Rogerio Feris, et al. Avlnet: Learning audio-visual language representations from instructional videos. *arXiv:2006.09199*, 2020.

[45] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clipfiltered 400 million image-text pairs. *arXiv:2111.02114*, 2021.

[46] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018.

[47] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019.

[48] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015.

[49] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, Sen Xing, Guo Chen, Junting Pan, Jiashuo Yu, Yali Wang, Limin Wang, and Yu Qiao. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022.

[50] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*, 2018.

[51] Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. Vlm: Task-agnostic video-language model pre-training for video understanding. *arXiv:2105.09996*, 2021.

[52] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *EMNLP*, 2021.

[53] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016.

[54] Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing high-resolution video-language representation with large-scale video transcriptions. In *CVPR*, 2022.

[55] Hongwei Xue, Yuchong Sun, Bei Liu, Jianlong Fu, Ruihua Song, Houqiang Li, and Jiebo Luo. Clip-vip: Adapting pre-trained image-text model to video-language representation alignment. *arXiv preprint arXiv:2209.06430*, 2022.

[56] Jinyu Yang, Jiali Duan, Son Tran, Yi Xu, Sampath Chanda, Liqun Chen, Belinda Zeng, Trishul Chilimbi, and Junzhou Huang. Vision-language pre-training with triple contrastive learning. In *CVPR*, 2022.

[57] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *arXiv:2111.07783*, 2021.

[58] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, Ce Liu, Mengchen Liu, Zicheng Liu, Yumao Lu, Yu Shi, Lijuan Wang, Jianfeng Wang, Bin Xiao, Zhen Xiao, Jianwei Yang, Michael Zeng, Luowei Zhou, and Pengchuan Zhang. Florence: A new foundation model for computer vision. *arXiv:2111.11432*, 2021.

[59] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Making visual representations matter in vision-language models. *CVPR 2021*, 2021.

[60] Ziqi Zhang, Zhongang Qi, Chunfeng Yuan, Ying Shan, Bing Li, Ying Deng, and Weiming Hu. Open-book video captioning with retrieve-copy-generate network. In *CVPR*, 2021.

[61] Ziqi Zhang, Yaya Shi, Chunfeng Yuan, Bing Li, Peijin Wang, Weiming Hu, and Zheng-Jun Zha. Object relational graph

with teacher-recommended learning for video captioning. In *CVPR*, 2020.

[62] Qi Zheng, Chaoyue Wang, and Dacheng Tao. Syntax-aware action targeting for video captioning. In *CVPR*, 2020.

[63] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018.

[64] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020.