

# Object-aware Video-language Pre-training for Retrieval

Alex Jinpeng Wang<sup>1</sup> Yixiao Ge<sup>2</sup> Guanyu Cai<sup>1,5</sup> Rui Yan<sup>1</sup> Xudong Lin<sup>4</sup>  
Ying Shan<sup>2</sup> Xiaohu Qie<sup>3</sup> Mike Zheng Shou<sup>1\*</sup>

<sup>1</sup>Show Lab, National University of Singapore <sup>2</sup>ARC Lab, <sup>3</sup>Tencent PCG

<sup>4</sup>Columbia University <sup>5</sup>Tongji University

## Abstract

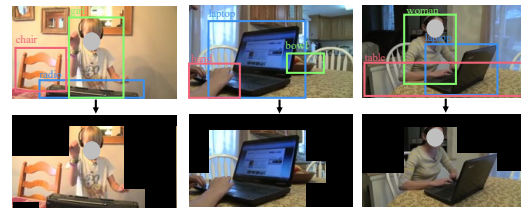
Recently, by introducing large-scale dataset and strong transformer network, video-language pre-training has shown great success especially for retrieval. Yet, existing video-language transformer models do not explicitly fine-grained semantic align. In this work, we present *Object-aware Transformers*, an object-centric approach that extends video-language transformer to incorporate object representations. The key idea is to leverage the bounding boxes and object tags to guide the training process. We evaluate our model on three standard sub-tasks of video-text matching on four widely used benchmarks. We also provide deep analysis and detailed ablation about the proposed method. We show clear improvement in performance across all tasks and datasets considered, demonstrating the value of a model that incorporates object representations into a video-language architecture. The code has been released in <https://github.com/FingerRec/OA-Transformer>.

## 1. Introduction

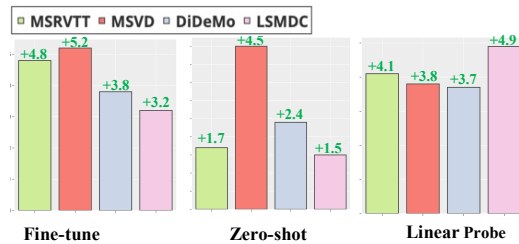
Learning scalable video-text representations for retrieval requires the understanding of both visual and textual clues, as well as the semantic alignment between these two modalities. Large-scale contrastive-based pre-training methods [4, 19] dominate the recent literature, where a “dual-encoder” framework (a video encoder and a text encoder) is trained in an end-to-end manner. Although these methods have led to great performance advances, *we figure out that the lack of regularization on fine-grained semantic associations hinders their further improvements.*

Thanks to the great progress of image-text pre-training [9, 21, 22, 24, 35, 37, 44], a series of methods attempt to leverage an off-the-shelf object detection model to generate richer information for cross-modality understanding, in-

Text: A little girl dancing to *music* and a teenage girl using a *computer*.



(a). The text and video can still be matched with object-guided masking.



(b). Performance gain for three downstream tasks over four benchmarks.

Figure 1. (a). **Masking object-irrelevant region keep the semantic unchanged.** From this example, we observe: 1. The object region is highly overlapped with visual salient region. 2. The predicted *Object Tags* has semantic relation with caption. e.g., *Music* and *Ratio*. *Laptop* and *Computer*. (b). **Our method vs. SOTA on three downstream tasks.** Motivated by (a), by incorporating the object into the learning of video-language pretraining with simple *Object-guided Masking*, we show promising results over multiple downstream video-language tasks.

cluding the visual objects and their tag concepts. The object information, together with the raw image and sentence, are then fed into a joint encoder for cross-modality interaction, leading to better correlations between regions and phrases. Given the success of object information in image-text pre-training, *it is intuitive to exploit the objects to improve video-text retrieval.* However, there exist some main challenges that prevent us from naïvely employing existing object-based techniques on video-text pre-training.

Fig. 1(a) shows that *object boxes and tags always focus on the salient regions and semantics*, which are considered

\*Corresponding Author.

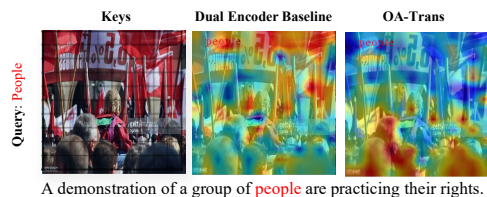


Figure 2. **Visualization of the cross-modality attention on a video-text sample.** This video is retrieved by the baseline dual encoder network [4] *wrongly* but *correctly* by our Object-aware Transformer (OA-Trans).

as the most important in each video. Existing object-based image-text pre-training methods either adopt an image-text joint encoder [21, 22] or cross-modality co-attention modules [24] for interaction between cross-modality local features. Despite the results being positive, it is impractical to adapt this paradigm from image domain to video domain. This is because all these methods require pre-extracted offline object feature for whole dataset. It would lead to **unaffordable computational overhead** to extract all objects, due to the billion-level frames. Moreover, their downstream performance heavily depends on the quality of the objects since they also need the objects as input for inference.

To this end, we introduce a simple yet effective paradigm for video-text pre-training, namely **Object-aware Transformer (OA-Trans)**, which explicitly enhances the fine-grained video-text interaction of the dominant “dual-encoder” framework at the same time maintaining its retrieval efficiency during inference. This is achieved by two novel designs in our method as follows.

(1) **Single anchor frame that encodes object information.** Instead of replacing all sampled video frames with their extracted object regions, we balance the matching recall and efficiency via combining whole frames together with a novel anchor frame that encodes object information. Specifically, we propose to only extract object regions on this anchor frame and softly mask out the non-object regions on this anchor frame.

(2) **A novel 4-stream object-aware contrastive (OAC) loss.** The input to our OA-Trans for pretraining include four stream: *raw video*, *anchor frame*, *object tags* (predicted object categories), and *raw text*. To explore how to combine these four streams, we do extensive experimental explorations and find out it works the best to contrast the *raw video* stream with the *object tags* stream and the *raw text* stream with the *anchor frame* stream. Note that the objects are only used for pre-training in our method, so the quality of detection has less effect on the downstream tasks and we do not need any extra computational overhead for downstream retrieval. As shown in Figure 2, a dual-network spreads its attention over the whole frame ran-

domly while OA-Trans with OAC loss can successfully focus on the “People” region.

Our contributions are as follows:

- We are the first to successfully develop an object-aware dual encoder model, namely OA-Trans, for end-to-end video-language pre-training.
- To alleviate the heavy cost of extracting object boxes, we propose to unify sampled whole frames with a single anchor frame whose non-object regions have been masked.
- We design a novel object-aware contrastive loss based on our unique input streams of video frames, textual query, the masked image, and predicted object tags on the anchor object frame.
- Our OA-Trans achieves significant improvements of Recall@1 on 4 benchmarks with three downstream tasks (Figure 1 (b)). *e.g.* MSVD (from 46.2% to 51.4%).

## 2. Related Work

### 2.1. Video-Language Pretraining

Limited by small-scale video-language datasets, previous video-language pretraining methods [12, 23, 26, 39, 41], have tended to use a combination of multiple “experts” to extract multi-modal features offline, *e.g.*, face, scene, object recognition action recognition, sound classification, and optical character recognition.

However, since a large-scale video-language dataset, HowTo100M [27], was proposed, there has been a trend of leveraging pretraining on large-scale data to learn better video-language representations. Most of these video-language pretraining methods [1, 20, 25, 29, 36] use a space-time CNN to pre-extract video features and propose a fusion module to align video features with language features that share the same semantics. Recently, considering most space-time CNNs are trained on Kinetics [14] that is much smaller than the pretraining dataset, to fully utilize massive information in pretraining datasets, end-to-end pretraining methods, ClipBert [19] and Frozen [4] are proposed.

### 2.2. Object in Vision-Language Tasks

Recently, object-centric models have been successfully applied in various vision-language tasks, such as visual question answering [2], image captioning [2], image-text retrieval [11, 18] and image-text pretraining [9, 21, 22, 24, 35, 37]. Especially in the field of image-text pretraining, since the proposal of Bottom-Up Top-Down attention (BUTD) [2], fine-grained features extracted from the level of objects gradually becomes the most common inputs of

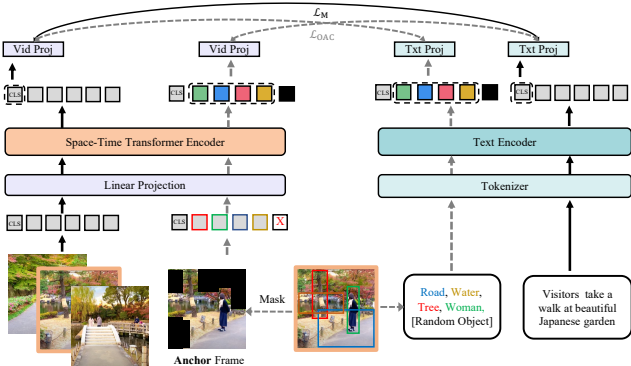


Figure 3. **Illustration of our Object-aware Transformer (OA-Trans).** The grey dotted line means the data flow are used only for pretrain but downstream tasks. The object tags and corresponding region guide the model to learn to attend to discriminative objects.

image-text pretraining models [9, 21, 22, 24, 35, 37]. Benefited from the object features that are the salient image regions and can be easily aligned with textual features, object-centric image-text models [9, 21, 22, 24, 35, 37] learn well-aligned image-text representations.

Although object-centric models have achieved remarkable progress in image-text pretraining, there lacks further exploration in video-text domain. ActBert [45] leverages object features to achieve better language-and-visual alignment. However, it needs to extract object features over the whole video and cooperate with features from other feature encoders. The computation cost of extracting object features and domain gap between different feature encoders prevent ActBert from an efficient and powerful object-centric model. Thus, how to appropriately bring object-level features to video-language pretraining still remains unsolved. In this work, to address the aforementioned issues, we propose an object-aware transformer to integrate object regions into video transformer [5] seamlessly.

### 3. Approach

The human visual system tends to focus on objects and other salient image regions [7, 30]. Representing video semantics using objects facilitates compositional semantic understanding because many perceptual components remain similar for a kind of object. Thus, a model that captures this compositional aspect potentially pays less attention to semantic-irrelevant information. Bringing this motivation into mind, we first revisit the current dual encoder framework in Section 3.1, which our model extends, and present Object-aware Transformer (OA-Trans) in Section 3.2. We further discuss the advantages of OA-Trans and different ways to utilize object information in Section 3.3.

### 3.1. Dual-encoder Framework

The previous works in video-language pretraining focus on aligning the raw-pixel video and raw text with a contrastive loss in both dual-encoder [4, 19] and one stream [42] frameworks. In this work, we choose the simple and effective dual-encoder framework (independent visual encoder and text encoder) Frozen [4] as our baseline. For the visual stream, a video project head is laid at the top of the visual encoder to project the output *cls* embedding into a shared embedding space. Similar to the visual stream, a text projection head is also laid at the top of the text encoder to project the *cls* token of text into shared embedding space. The same as text stream and the normalized embedding of video and text is recorded as  $v$  and  $t$ , respectively.

**Objective:** To train this dual-encoder framework, the normalized embedding of matched text-video pairs in the batch are treated as positives, and all other pairwise combinations in the batch are treated as negatives. In practice, supposing we have  $K$  samples in a batch, then the symmetrical contrastive loss is introduced as follow:

$$\mathcal{L}_{v2t} = -\log \frac{\exp(\text{sim}(v, t)) / \tau}{\sum_{i=0}^K \exp(\text{sim}(v^i, t))} \quad (1)$$

$$\mathcal{L}_{t2v} = -\log \frac{\exp(\text{sim}(t, v)) / \tau}{\sum_{i=0}^K \exp(\text{sim}(t^i, v))}, \quad (2)$$

where  $\tau$  is the temperature and  $\text{sim}$  is a similarity function (*i.e.*, dot product). The final video-text matching loss is  $\mathcal{L}_M = \mathcal{L}_{v2t} + \mathcal{L}_{t2v}$ .

### 3.2. Object-aware Transformer

In this section, we present our efficient and simple Object-aware Transformer (OA-Trans) in detail. The pipeline of OA-Trans is shown in Fig. 3. The distinction from the baseline is the additional masked image stream and object tag stream. Given an input pair of video and text, we first sample one video clip from this video. Then we find the central index from this clip and find the closest object frame. From this object frame, we generate the masked **an-chor** object image and object tags.

Instead of using *cls* token, we average tokens from non-masked patches for the masked image and the normalized embedding is represented as  $v_l$ . Similarly, the output for the object tag stream is represented as  $t_l$ . Then we compute the matching loss  $\mathcal{L}_M$  and Object-aware Contrastive (OAC) loss  $\mathcal{L}_{OAC}$  from their corresponding output. Next, we introduce the key components and their design motivation of this pipeline as below:

Method	Years	Vis Enc. Init.	Pretrained Data	R@1	R@5	R@10	MedR
ActBERT [45]	CVPR'20	VisGenome	[136M] HowTo100M	16.3	42.8	56.9	10.0
VidTranslate [16]	Arxiv'20	IG65M	[136M] HowTo100M	14.7	-	52.8	
NE [1]	AAAI'21	ImageNet, Kinetics	[136M] HowTo100M	17.4	41.6	53.6	8.0
ClipBERT [19]	ICCV'21	-	[5.6M] COCO, VisGenome	22.0	46.8	59.9	6.0
MMT [12]	ECCV'20	Numerous experts	[136M] HowTo100M	26.6	57.1	69.6	4.0
Frozen [4]	ICCV'21	ImageNet	[3M] CC3M	25.5	54.5	66.1	4.0
Frozen [4]	ICCV'21	ImageNet	[5.5M] CC3M, WebVid-2M	31.0	59.5	70.5	3.0
Frozen[Our Imp.]	ICCV'21	ImageNet	[5.5M] CC3M, WebVid-2M	33.2	61.5	71.9	3.0
Support Set [29]	ICLR'21	IG65M, ImageNet	[136M] HowTo100M	30.1	58.5	69.3	3.0
<b>OA-Trans</b>		ImageNet	[2.5M] Webvid-2M	<b>32.7</b>	<b>60.9</b>	<b>72.5</b>	<b>3.0</b>
<b>OA-Trans</b>		ImageNet	[5.5M] CC3M, WebVid-2M	<b>35.8</b>	<b>63.4</b>	<b>76.5</b>	<b>3.0</b>
OA-Trans $\ddagger$		CLIP-WIT	[5.5M] CC3M, WebVid-2M	39.4	68.8	78.3	2.0
OA-Trans $\ddagger$ [12F]		CLIP-WIT	[5.5M] CC3M, WebVid-2M	40.9	70.4	80.3	2.0
<i>Zero-shot</i>							
HT MIL-NCE [27]	CVPR'20	-	[136M] HowTo100M	7.5	21.2	29.6	38.0
SupportSet [29]	ICLR'21	IG65M, ImageNet	[136M] HowTo100M	8.7	23.0	31.1	31.0
Frozen [4]	ICCV'21	ImageNet	[2.5M] WebVid-2M	14.5	29.5	64.5	21.0
Frozen [4]	ICCV'21	ImageNet	[5.5M] CC3M, WebVid-2M	18.7	39.5	51.6	10.0
Frozen [4] [Our Imp.]	ICCV'21	ImageNet	[5.5M] CC3M, WebVid-2M	21.7	45.5	53.9	9.0
CLIP[12F] [31]	Arxiv'21	CLIP-WIT	-	28.5	49.7	61.2	5.0
<b>OA-Trans</b>		ImageNet	[2.5M] Web Vid-2M	<b>18.4</b>	<b>36.5</b>	<b>46.8</b>	<b>10.0</b>
<b>OA-Trans</b>		ImageNet	[5.5M] CC3M, WebVid-2M	<b>23.4</b>	<b>47.5</b>	<b>55.6</b>	<b>8.0</b>
OA-Trans $\ddagger$		CLIP-WIT	[5.5M] CC3M, WebVid-2M	29.7	52.1	63.5	5.0
OA-Trans $\ddagger$ [12F] $\ddagger$		CLIP-WIT	[5.5M] CC3M, WebVid-2M	31.4	55.3	64.8	4.0

Table 1. Comparison with state-of-the-art results on MSRVT for text-to-video retrieval.  $\ddagger$  denotes the model is initialized with weights from CLIP [31]. **Vis Enc. Init.:** Datasets that visual encoders’ initial weights are trained on.

**Anchor Object Frame.** Given a video with arbitrary length, we first uniform sample  $L$  (*i.e.*,  $L = 8$ ) frames, and an improved 1600-class Faster RCNN [2] is used to extract  $N$  objects offline (probably over-sampled and noisy). We save these offline objects on a disk for reuse. During training, we select top- $K$  objects with unique object categories. If the object is too large, we reduce the object size to half of it. The analysis of the object number is provided in Sec. 4.8.

**Masking.** Given an object frame with  $N$  object regions, we first *mask* the region that does not contain objects. We then divide an *masked* frame into regular non-overlapping patches. Then we sample a subset of patches that contain the object region and mask the remaining ones to form a regular grid. In this way, a patch will be either masked out or remain its actual pixels. To prevent overfitting, we drop 20% objects randomly and shift the *anchor* frame to an adjacent frame in time. In addition, we crop the central region if the object region is too large. We simply refer to this as ”object-guided masking”. With the proposed Object-Guided Masking, the model is forced to learn to understand the context information and relationships of objects, rather than simply modeling scene bias.

**Object-aware Contrastive (OAC) Loss.** Since our aim is to enhance the fine-grained representation, the straightforward idea is to align the predicted object tags and the local masked image directly. However, this naive approach will not be able to directly benefit downstream applications

because objects are not input to the model for downstream applications. And the loss is quite easy to optimize and may fail into trivial solutions and further damage the learning of global video to text matching. Based on this observation, we propose a novel OAC loss with cross guidance from object regions to captions and from object tags to video frames.

Specifically, we first use object tags to align with raw video. Formally,

$$\mathcal{L}_{tag} = -\log \frac{\exp(\text{sim}(v, t_l) / \tau)}{\sum_{i=0}^K \exp(\text{sim}(v^i, t_l) / \tau)} \quad (3)$$

Although the object tags are from a limited 1600-class dictionary defined by Visual Genome [17], the tags are usually capable of capturing relevant high-level semantics presented in captions. For example, *Woman* and *Visitors*, *Tree* and *Garden* in the Fig. 3. Then if we encourage the global visual embedding  $v$  not only to align with  $t$  but also  $t_l$ , the model will strengthen the association between different nouns potentially.

Similarly, we force the model to align the full sentence with a masked object frame. Formally,

$$\mathcal{L}_{mask} = -\log \frac{\exp(\text{sim}(v_l, t) / \tau)}{\sum_{i=0}^K \exp(\text{sim}(v_l, t^i) / \tau)} \quad (4)$$

Combining these complementary cross guidance, we define the OAC Loss as:  $\mathcal{L}_{OAC} = \mathcal{L}_{tag} + \mathcal{L}_{mask}$

**Overall Training Objective.** The final loss function of OA-Transformer is:

$$\mathcal{L} = \mathcal{L}_M + \lambda \mathcal{L}_{OAC}, \quad (5)$$

where  $\lambda$  is the coefficient that controls the balance between global match loss and OAC loss.

By forcing both video encoder and text encoder to mine object-centric information, our video-text model directly benefits from the high-level semantics captured by object regions and object tags. As a result, the OA-Trans learns more discriminative representations for downstream video-text tasks.

### 3.3. Discussion

**Advantages.** There exist several advantages for the OA-Trans: *i.* We only use one object image as a reference during pretraining and the additional computation cost is limited. *ii.* The object knowledge is learned during pretraining, thus reducing the effects of noisy objects on downstream tasks. *iii.* Our paradigm without the requirement of modifying the architecture of base vision encoder that can be plug-and-play into existing video-language pretraining methods.

**More Ways to Incorporate Objects.** Besides the simple masking operation, we also empirically studied multiple ways to utilize objects in both vision and language modality inspired by previous works [15, 22]. For visual modality, we consider *Pure Offline Features* and *The joint modeling of Offline Feature with Raw-pixel Video*. All these design details are presented in the supplementary. We compare all design choices and show our solution is the superior design.

## 4. Experiments

We evaluate our Object-aware Transformer (OA-Trans) on several video-text benchmarks. Specifically, we consider the following tasks: Video-Text Retrieval (Section 4.4) and Linear Probe Evaluation (Section 4.5).

### 4.1. Pretraining Datasets

Since the widely-used dataset, i.e., HowTo100M [27], is heavily noisy and only contains instructional videos. In this work, we adopt two clean datasets: *(i)* WebVid2.5M (video-text); and *(ii)* Google Conceptual Captions (image-text) to cover more generalized scenarios.

**WebVid2.5M** [4] consists of 2.5M video-text pairs, which is an open domain video captioning dataset. The manually generated captions are well-formed sentences. **Google Conceptual Captions (CC3M)** is scraped from the web and more than 10% of CC3M images are in fact thumbnails from videos. As some images are missing on the web, we get 2.97M images in total.

Method	R@1	R@5	R@10	MedR
<i>Sentence to Video</i>				
Multi. Cues [28]	20.3	47.8	61.1	6.0
CE [23]	19.8	49.0	63.8	6.0
Support Set [29] (HowTo PT)	28.4	60.0	72.9	4.0
Forzen [4]	33.7	64.7	76.3	3.0
<b>OA-Trans</b>	<b>39.1</b>	<b>68.4</b>	<b>80.3</b>	<b>2.0</b>
<i>Multiple Sentences to Video</i>				
TeacherText [10]	25.4	56.9	71.3	4.0
<b>OA-Trans</b>	<b>51.4</b>	<b>82.3</b>	<b>88.0</b>	<b>2.0</b>

Table 2. Text-to-video retrieval results on MSVD [8].

### 4.2. Downstream Datasets

To verify the effectiveness of learned visual and textual representations, we evaluate OA-Trans on four video-text benchmarks as follows:

**MSRVTT** [40] contains 10K YouTube videos with 200K descriptions. Following the previous works [?, 4], we use 9K videos for training and report results on the 1K test set.

**DiDeMo** [3] contains 10K Flickr videos. Each video is annotated with multiple captions, which results in 40K sentences in total. In the experiments, all captions of a video are regarded as a single description.

**MSVD** [8] contains 20K YouTube videos annotated with 100K sentences. The training set contains 10K videos, and we report results on the validation set with 4.9K videos. Since each video is annotated with multiple sentences, we report both *Sentence to Video* and *Multiple Sentences to Video* results to compare with related works.

**LSMDC** [32] contains 12K video-text pairs from 202 movies. Following [33], the validation set contains 7K pairs, and evaluation is conducted 1K test set.

### 4.3. Setup

**Backbone.** The main components of our method are Visual Encoder and Textual Encoder. For the Textual Encoder, we adopt Distill Bert [34] as default. For the Visual Encoder, we adopt Vision Transformer with space-time attention from TimeSformer [6]. For the Vision Transformer, the 12-layer ViT-B/16 is used as the backbone. All models trained for 128 epochs.

**Technical Detail.** We use the Adam optimizer with weight decay regularization and decay the learning rate with a cosine schedule. When pretraining on WebVid2.5M, 1 object reference frame and 4 video frames are sampled. For CC3M, the video frame number is set to 1 because CC3M is an image-text dataset. The control weight  $\lambda$  is set to 0.5 experimentally.

The whole pretraining takes 5 days on 64 Tesla A100 GPUs. Unless otherwise specified, all results reported in this paper adopt the best model. When fine-tuning the pretrained model, only 8 video frames are sampled on all downstream tasks.

Method	R@1	R@5	R@10	MedR
S2VT [38]	11.9	33.6	-	13.0
FSE [43]	13.9	36.0	-	11.0
CE [23]	16.1	41.1	-	8.3
ClipBERT [19]	20.4	44.5	56.7	7.0
Frozen [4]	31.0	59.8	72.4	3.0
<b>OA-Trans</b>	<b>34.8</b>	<b>64.4</b>	<b>75.1</b>	<b>3.0</b>
<i>Zero-shot</i>				
Frozen [4]	21.1	46.0	56.2	7.0
<b>OA-Trans</b>	<b>23.5</b>	<b>50.4</b>	<b>59.8</b>	<b>6.0</b>

Table 3. Text-to-video retrieval results on DiDeMo. We show both the fine-tune and zero-shot retrieval results .

#### 4.4. Video-Text Retrieval

**MSRVTT.** Table 1 summarizes the results on **MSRVTT**. Besides ClipBERT and Support Set, other methods are pretrained on 136M clip-caption pairs from HowTo100M. To ensure a fair comparison, we re-implement the previous SOTA method, Frozen [4], with a distributed parallel training schedule. Under the full fair comparison, OA-Trans outperforms the previous best method Frozen by 2.6% on R@1. Surprisingly, only pretrained with open-domain 2.5M video-text pairs, our method already outperforms all previous works that are pretrained on 136M clip-caption pairs.

Typically, to evaluate the generalization of models, we also report zero-shot results, i.e., no fine-tuning is conducted. Our method outperforms previous methods significantly. The results show that our model has a better generalization ability than others. To further verify our method can extend to strong visual backbones, we initialize the visual encoder with CLIP’s weights [31]. As the results shown in Table 1, our method still improves the performance of CLIP. Thus, our method works well with different initial weights even if the loaded initial weights already have a strong performance.

**MSVD.** Because each video is annotated with multiple captions, previous works are mainly divided into two types: *i.* Sentence to video: Treat each sentence as the textual query. *ii.* Multiple sentences to video: Combine multiple sentences of a video as the textual query. The results are shown in Table 2, in both settings, our method outperforms other methods by 5% on R@1 at least.

We also show the retrieval results on **DiDeMo** and **LSMDC** in Table 3 and Table 4. OA-Trans outperforms previous methods on all metrics.

#### 4.5. Linear Probe Evaluation

The linear probe is an important measurement to evaluate the quality of representations learned in large-scale image-text pretraining [31] and image self-supervised pretraining [13]. However, this technique is never explored in video-text pretraining and most related works still focus on

Method	R@1	R@5	R@10	MedR
JSFusion [41]	9.1	21.2	34.1	36.0
MEE [26]	9.3	25.1	33.4	27.0
CE [23]	11.2	26.9	34.8	25.3
Forzen [4]	15.0	30.8	39.8	20.0
MMT (HowTo PT) [12]	12.9	29.2	38.8	19.3
<b>OA-Trans</b>	<b>18.2</b>	<b>34.3</b>	<b>43.7</b>	<b>18.5</b>

Table 4. Text-to-video retrieval results on LSMDC.

Method	VE Init.	MSR	MSVD	DiDeMo	LSMDC
Frozen [4]	ImageNet	27.2	30.3	26.6	13.2
<b>OA-Trans</b>	ImageNet	<b>31.3</b>	<b>34.1</b>	<b>30.4</b>	<b>18.1</b>
Clip [31]	CLIP-WIT	30.5	34.5	29.8	16.8
OA-Trans ‡	CLIP-WIT	33.2	36.9	34.8	21.5

Table 5. The linear probe evaluation of three video-text retrieval datasets. ‡ means we use CLIP weight for visual encoder initialization. We report R@1 result and VE Init is short for Visual Encode Initialization.

fine-tuning the overall model.

The fine-tune strategy brings two problems: *i.* The hyper-parameter spaces for various downstream datasets are very large. It’s very difficult to provide fair comparisons among different pretrain methods. *ii.* Fine-tuning adjusts the overall model and adapts representations to a specific dataset, it may hide the failures that a model does not learn general and robust representations.

Following CLIP [31], in this work we fit a linear classifier on representations extracted from the pretrained model and measure its performance on various downstream datasets. We implement Frozen and CLIP by ourselves. Since CLIP is an image-text pretrain method, we sample 8 frames of each video and average the image-level feature to represent a video. The results are shown in Table 5. We also show the results of OA-Trans initialized with CLIP-pretrained weights. It can be seen that OA-Trans generalizes well to these datasets. We hope this experiment will inspire the community to focus more on this task.

#### 4.6. Qualitative Visualization

**Attention Region Visualization.** To provide insight into the inner representation of OA-Trans, we provide further visualization. Specifically, we visualize the attention map between captions and visual patches, where a text token is regarded as the query and attention weights on all spatial tokens are visualized. We use the output of the first Transformer layer for visualization. To analyze if OA-Trans only helps the modeling of nouns that are included in the object tag dictionary. We select nouns from both the object tag dictionary and other novel object tags that are not included in the object tag dictionary.

The visualization of the attention weights allocated to each patch is shown in Fig. 4 and we make the following observations: *i.* For the complex scenarios like “awards held”

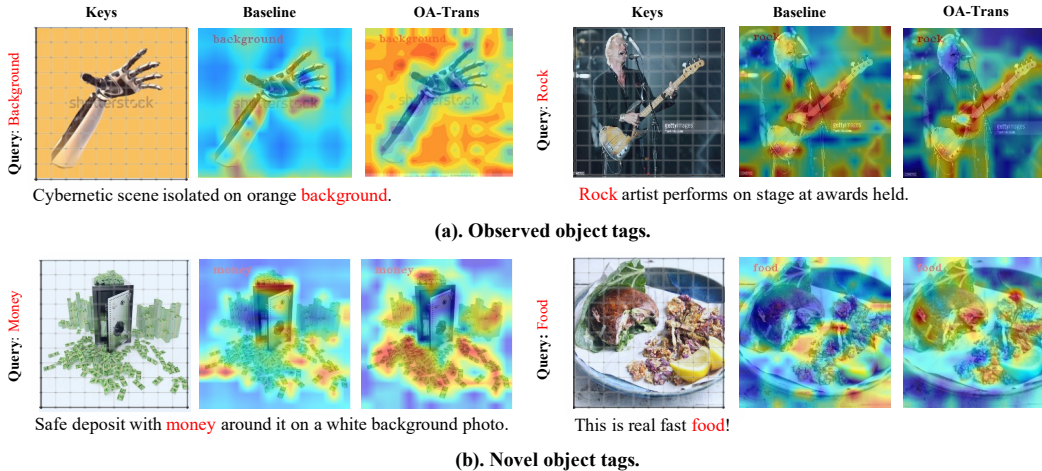


Figure 4. **Cross-modality attention regions visualization.** The specific text token as query and the patch-level tokens as keys. In the upper part, “people” and “rock” are in the predefined object vocabulary. In the bottom part, “food” and “money” are not covered by the predefined object vocabulary.

$\mathcal{L}_{tag}$	$\mathcal{L}_{mask}$	T2V			V2T		
		R@1	R@5	R@10	R@1	R@5	R@10
		14.5	31.6	40.8	14.8	29.7	40.6
✓		17.4	33.2	45.7	18.1	33.6	42.7
	✓	15.9	33.2	43.3	15.4	30.9	40.8
✓	✓	18.4	36.2	47.8	17.5	33.0	46.4

Table 6. The ablation of object category and object region on MSRVT.  $\mathcal{L}_{tag}$  means object tags to video match loss and  $\mathcal{L}_{mask}$  means object mask image to text match loss.

in the up-right of Fig. 4, OA-Trans focuses on rock devices more accurately while baseline looks at irrelevant corners. *ii.* Interestingly, even “money” and “food” are not included in the object tag dictionary, OA-Trans still focuses on the corresponding regions accurately. *This experiment demonstrates the introduction of object tags and regions improves the overall representation ability rather than fits an implicit bias over object tags.*

#### 4.7. Efficiency Analysis

Since we only use two normal stream without object when retrieval on downstream datasets, our OA-Trans has a very fast inference running time even for retrieval on million-scale datasets. We use the popular similarity search/ranking library FAISS-GPU<sup>1</sup> on a server with 8 A100 GPUs and 88 Kernel Intel(R) Xeon(R) Platinum 8255C CPU @ 2.50GHz. Given a new query, Table 7 below shows the time needed for visual encoding, textual encoding, and similarity ranking (1st row for thousand-scale and 2nd row for million-scale). Given a new query, the total search time on HowTo100M is (5.88+143.25=149.13)ms

<sup>1</sup><https://github.com/facebookresearch/faiss>

for text-to-video retrieval and (23.92+143.25=167.17)ms for video-to-text retrieval, which is acceptable in practice.

Dataset	# Samples	VE	TE	Ranking
MSR	1K	23.92ms	5.88ms	1.51ms
HowTo100M	128.94M	23.92ms	5.88ms	143.25ms

Table 7. Running time analysis for OA-Trans during retrieval/inference. Numbers are averaged over 1000 runs. VE and TE means Visual Encoder and Text Encoder, correspondingly.

#### 4.8. Ablation Studies

In this section, we conduct ablation studies and analyze the different choices for utilizing object information. We pretrain our OA-Trans on WebVid2.5M and conduct an evaluation on zero-shot MSRVT retrieval.

**Effectiveness of Each Component.** In this section, we explore the effect of object region and object tag. The results are given in Table 6. When using object tag, our method achieves 1.4% R@1 gain compared to the baseline in text-to-video retrieval. We also find object tags contribute more to the retrieval ability. The combination of object tag and object region leads to the best result.

**Number of Objects.** In the left of Fig. 5, we compare the results of different OA-Trans by varying the number of objects. We find that more objects lead to better performance in general. When the number of objects is larger than 10, the performance remains consistent. Thus, the number of objects is set to 10 as default.

We also explore the impact of the mask patch probability in the right of Fig. 5. For this experiment, we take mask

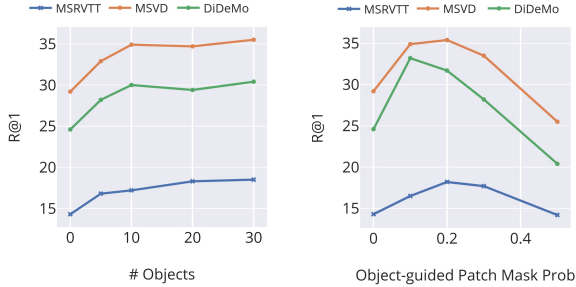


Figure 5. Left: The number of object and the corresponding Retrieval top-1 result. Right: The object-guided mask probability and the corresponding R@1 result.

Method	T2V			V2T		
	R@1	R@5	R@10	R@1	R@5	R@10
Baseline	14.5	31.6	40.8	14.8	29.7	40.6
Padding [22]	15.5	33.2	43.4	15.7	30.5	41.2
Two Stream†	17.5	35.9	47.8	17.5	35.7	46.5
Two Stream	17.7	35.5	48.1	18.2	34.7	45.6

Table 8. **The variations of utilizing object categories.** Two Stream† means Two Stream + Padding.

probability from 0 to 0.5 for comparison. We can see that the accuracy grows firstly as the probability increases for all three datasets. But when the probability is larger than 0.2, all results drop significantly. The large mask probability will drop too many regions and the semantic may change.

**Strategy of Object Tag Utilization.** In this section, we investigate the different ways to utilize the object tag. We study three variations: *i. Padding:* Pad object tags to the original caption as in Oscar [22]. *ii. Two Stream:* Use two-stream input. One stream is the original caption, the other stream is the object tags. *iii. Two Stream + Padding:* Use two-stream input. One stream is the original caption, the other stream is the original caption with padding object tags. Notice all the strategies are designed for pretraining. During testing, we use normal video-text retrieval settings to show the generalization of our method.

The results are shown in Table 8. We find padding operation leads to around 1% improvement on both text-to-video and video-to-text retrieval settings. The reason behind this phenomenon is that the padding operation performs like an augmentation to the text. When introducing a two-stream pipeline, we find the R@1 for both text-to-video and video-to-text tasks is improved by around 3%. In such a form, the model is asked not only to align a video with its original caption but also the padding of detailed objects. Thus, object information that is not mentioned in the caption is also preserved in the visual representation. Such visual representations could help the pretrained model generalize well to more scenarios. In this work, we adopt the Two Stream

Method	R@1	R@5	R@10	MedR
Mask Only	16.4	35.5	45.8	11.0
Raw Video Only	15.9	33.2	43.3	12.0
Joint Input	18.5	37.2	49.8	10.0

Table 9. **The comparison of alternative inputs in visual stream.** We report the zero-shot retrieval result.

strategy as default.

**Alternative Inputs in Visual Stream.** In this section, we give a comparison between different visual inputs to see which one helps to capture better representations in our OA-Trans. Specifically, we keep other components unchanged and then we compare three visual inputs as follows: *i. Raw Video Input:* Only input the original video. *ii. Only Masked Input:* We remove the original raw video stream and only input masked anchor image. *iii. Joint Input:* Input the masked anchor image and the raw video stream.

The results are reported in Table 9. Interestingly, we find the Mask Only input already suppresses Raw Video Only around 2.5% over R@10 metric. This demonstrates the importance of object-centric modeling in video-text matching. Compared with single-stream input, the joint input leads to the best result. This Phenomenon indicates that these two streams provide complementary information and the model can benefit from object-region guided local alignment.

## 5. Conclusion

Current dual-encoder networks in video-language pre-training lack the learning of fine-grained semantic alignment. Objects can provide a strong complement for this problem, but their modeling is very challenging for machine vision especially in video. The OA-Trans we present here makes use of a simple object bounding box and object tags information to generate a contextualized representation of the entire scene. We note that such integration is particularly natural in cross-modality transformer models, where an object region has the same role in the architecture as the uniformly-spaced patch tokens.

In our current implementation, we use an externally provided offline object detector. However, it will be interesting to replace the offline bounding boxes with boxes that the model generates itself without strong supervision. An additional interesting extension is to cluster visual similar regions in an video in a self-supervised fashion, where the task is to align the clustered video with text. We leave these challenges to future work.

## Acknowledgement

This project is supported by the National Research Foundation, Singapore under its NRFF Award NRF-NRFF13-2021-0008 and Mike Shou’s Start-Up Grant from NUS.



## References

- [1] Elad Amrani, Rami Ben Ari, Daniel Rotman, and Alex Bronstein. Noise estimation using density estimation for self-supervised multimodal learning. *arXiv preprint arXiv:2003.03186*, 2020. 2, 4
- [2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018. 2, 4
- [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. 5
- [4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. *arXiv preprint arXiv:2104.00650*, 2021. 1, 2, 3, 4, 5, 6
- [5] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv:2102.05095*, 2021. 3
- [6] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*, 2021. 5
- [7] Linda L Chao and Alex Martin. Representation of manipulable man-made objects in the dorsal stream. *Neuroimage*, 12(4):478–484, 2000. 3
- [8] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 190–200, 2011. 5
- [9] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. UNITER: Universal image-text representation learning, 2020. 1, 2, 3
- [10] I. Croitoru, S. Bogolin, M. Leordeanu, H. Jin, A. Zisserman, S. Albanie, and Y. Liu. Teactext: Crossmodal generalized distillation for text-video retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11583–11593, 2021. 5
- [11] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612*, 2017. 2
- [12] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020. 2, 4, 6
- [13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 6
- [14] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The Kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017. 2
- [15] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. *ICML*, 2021. 5
- [16] Bruno Korbar, Fabio Petroni, Rohit Girdhar, and Lorenzo Torresani. Video understanding as machine translation. *arXiv preprint arXiv:2006.07203*, 2020. 4
- [17] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017. 4
- [18] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching, 2018. 2
- [19] Jie Lei, Linjie Li, Luwei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: ClipBERT for video-and-language learning via sparse sampling. *CVPR*, 2021. 1, 2, 3, 4, 6
- [20] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. *EMNLP*, 2020. 2
- [21] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 1, 2, 3
- [22] Xiujuan Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*, 2020. 1, 2, 3, 5, 8
- [23] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. In *BMVC*, 2019. 2, 5, 6
- [24] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NIPS*, 2019. 1, 2, 3
- [25] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Xilin Chen, and Ming Zhou. UniVL: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020. 2
- [26] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a text-video embedding from incomplete and heterogeneous data. *arXiv*, 2018. 2, 6
- [27] 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. 2, 4, 5
- [28] Niluthpol Chowdhury Mithun, Juncheng Li, Florian Metze, and Amit K Roy-Chowdhury. Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 19–27, 2018. 5
- [29] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander Hauptmann, João Henriques, and Andrea

- Vedaldi. Support-set bottlenecks for video-text representation learning. *arXiv preprint arXiv:2010.02824*, 2020. 2, 4, 5
- [30] Anthony Quinton. Objects and events. *Mind*, 88(350):197–214, 1979. 3
- [31] 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. *arXiv preprint arXiv:2103.00020*, 2021. 4, 6
- [32] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *CVPR*, 2015. 5
- [33] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. Movie description. *International Journal of Computer Vision*, 123(1):94–120, 2017. 5
- [34] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019. 5
- [35] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vi-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*, 2020. 1, 2, 3
- [36] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019. 2
- [37] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019. 1, 2, 3
- [38] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729*, 2014. 6
- [39] Wenzhe Wang, Mengdan Zhang, Runnan Chen, Guanyu Cai, Penghao Zhou, Pai Peng, Xiaowei Guo, Jian Wu, and Xing Sun. Dig into multi-modal cues for video retrieval with hierarchical alignment. In *IJCAI*, 2021. 2
- [40] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 5
- [41] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018. 2, 6
- [42] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. *arXiv preprint arXiv:2106.02636*, 2021. 3
- [43] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *ECCV*, 2018. 6
- [44] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588, 2021. 1
- [45] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020. 3, 4