Object-aware Video-language Pre-training for Retrieval

Alex Jinpeng Wang¹ Yixiao Ge² Guanyu Cai¹,⁵ Rui Yan¹ Xudong Lin⁴
Ying Shan² Xiaohu Qie³ Mike Zheng Shou¹*

¹Show Lab, National University of Singapore  ²ARC Lab,³Tencent PCG
⁴Columbia University  ⁵Tongji 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-
gained semantic align. In this work, we present Object-
aware Transformers, an object-centric approach that ex-
tends 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 pro-
posed 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 re-
leased 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 modal-
ities. Large-scale contrastive-based pre-training meth-
ods [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 associa-
tions 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 gener-
ate richer information for cross-modality understanding, in-
cluding the visual objects and their tag concepts. The ob-
ject information, together with the raw image and sentence,
are then fed into a joint encoder for cross-modality inter-
action, 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 im-
prove video-text retrieval. However, there exist some main
challenges that prevent us from naively 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 1. (a). Masking object-irrelevant region keep the se-


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

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

Text: A little girl dancing to music and a teenage girl using a computer.
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 randomly 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 [1-3] 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
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:

\[ L_{v2t} = -\log \frac{\exp(sim(v, t))/\tau}{\sum_{i=0}^{K} \exp(sim(v^i, t))} \]

(1)

\[ L_{t2v} = -\log \frac{\exp(sim(t, v))/\tau}{\sum_{i=0}^{K} \exp(sim(t^i, v))} \]

(2)

where τ is the temperature and sim is a similarity function (i.e., dot product). The final video-text matching loss is \( L_M = L_{v2t} + L_{t2v} \).

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 anchor 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 \( L_M \) and Object-aware Contrastive (OAC) loss \( L_{OAC} \) from their corresponding output. Next, we introduce the key components and their design motivation of this pipeline as below:
Combining these complementary cross guidance, we define the OAC Loss as: 
\[ L_{OAC} = L_{tag} + L_{mask} \]
Overall Training Objective. The final loss function of OA-Transformer is:
\[
\mathcal{L} = \mathcal{L}_M + \lambda \mathcal{L}_{OAC},
\]
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: 

1. We only use one object image as a reference during pretraining and the additional computation cost is limited.

2. The object knowledge is learned during pretraining, thus reducing the effects of noisy objects on downstream tasks.

3. 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.

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:

- **MSRVT** [40] contains 10K YouTube videos with 200K descriptions. Following the previous works [2,4], 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.
Table 3. Text-to-video retrieval results on DiDeMo. We show both the fine-tune and zero-shot retrieval results.

| 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    | 20.4 | 44.5 | 56.7 | 7.0  |
| Frozen [4]  | 31.0 | 59.8 | 72.4 | 3.0  |
| OA-Trans    | 34.8 | 64.4 | 75.1 | 3.0  |

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  |
| OA-Trans‡  | ImageNet| 31.3| 34.1 | 30.4   | 18.1  |
| 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.

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 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”...
in the up-right of Fig. 4, OA-Trans focuses on rock devices more accurately while baseline looks at irrelevant corners. 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 ¹ 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 for text-to-video retrieval and (23.92+143.25=167.17)ms for video-to-text retrieval, which is acceptable in practice.

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 MSR VTT 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

¹https://github.com/facebookresearch/faiss

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 |
|---------------------|---------------------|-----|-----|
|                     | $R@1$               | $R@5$ | $R@10$ | $R@5$ | $R@10$ |
| ✓                   | 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 MSRVTT. $\mathcal{L}_{tag}$ means object tags to video match loss and $\mathcal{L}_{mask}$ means object mask image to text match loss.

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

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 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 OATrans. 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 pretraining 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 OATrans 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-paced 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, Güll Väröl, 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. Teachtext: 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] Valentijn 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, Luowei 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, 5, 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] Xiujun 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, 4, 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] Huaiashao 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] Nilothpol 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

3311
[30] Anthony Quinton. Objects and events. *Mind*, 88(350):197–214, 1979.
[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.
[32] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *CVPR*, 2015.
[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.
[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.
[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.
[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.
[37] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019.
[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.
[39] Wenzhe Wang, Mengdan Zhang, Runnan Chen, Guanyu Cai, Penghao Zhou, Pui Peng, Xiaowei Guo, Jian Wu, and Xing Sun. Dig into multi-modal cues for video retrieval with hierarchical alignment. In *IJCAI*, 2021.
[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.
[41] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018.
[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.
[43] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *ECCV*, 2018.
[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.
[45] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020.