Masking Modalities for Cross-modal Video Retrieval

Valentin Gabeur[1,2] Arsha Nagrani[2] Chen Sun[2] Karteek Alahari[1] Cordelia Schmid[2]
[1] Inria* [2] Google Research

Abstract

Pre-training on large scale unlabelled datasets has shown impressive performance improvements in the fields of computer vision and natural language processing. Given the advent of large-scale instructional video datasets, a common strategy for pre-training video encoders is to use the accompanying speech as weak supervision. However, as speech is used to supervise the pre-training, it is never seen by the video encoder, which does not learn to process that modality. We address this drawback of current pre-training methods, which fail to exploit the rich cues in spoken language. Our proposal is to pre-train a video encoder using all the available video modalities as supervision, namely, appearance, sound, and transcribed speech. We mask an entire modality in the input and predict it using the other two modalities. This encourages each modality to collaborate with the others, and our video encoder learns to process appearance and audio as well as speech. We show the superior performance of our 'modality masking' pre-training approach for video retrieval on the How2R, YouCook2 and Condensed Movies datasets.

1. Introduction

We live in a multimodal world, communicating through speech, visual signals and sound. This is reflected in the videos created and uploaded online—often they are accompanied by a highly informative audio track containing cues complementary to visual content. Our goal in this work is to perform video retrieval with natural language queries.

While many popular video understanding works [4, 15, 18, 22, 27, 36] restrain the video signal to a sequence of visual frames, several approaches [9, 20, 23] have progressed to incorporate information from different modalities through the use of pre-trained feature extractors called "experts". For videos, naturally composed of multimodal information, learning the optimal fusion of different modality 'experts' is paramount. This challenge of multimodal learning is made more difficult by the scarcity of large manually-captioned video datasets. Existing datasets, e.g., [16, 19, 30, 36, 40] remain small scale. This has led to a several approaches utilising the large amount of instructional videos online [9, 22, 24, 27], where transcribed speech (obtained with ASR) is closely linked to visual content, and hence a valuable source of supervision to train video encoders. Because of the proximity between text queries and speech, this approach presents the advantage of transferring well to text-to-video retrieval tasks. However, because the speech modality is used as a source of 'pseudo' captioning labels, most of these works [9, 22, 24] only pre-train an encoder to process non-speech modalities (RGB, audio, etc), thereby not learning to combine speech and visual inputs effectively during pre-training. For many videos online, effectively processing speech is crucial for accurate video retrieval (Fig. 1).

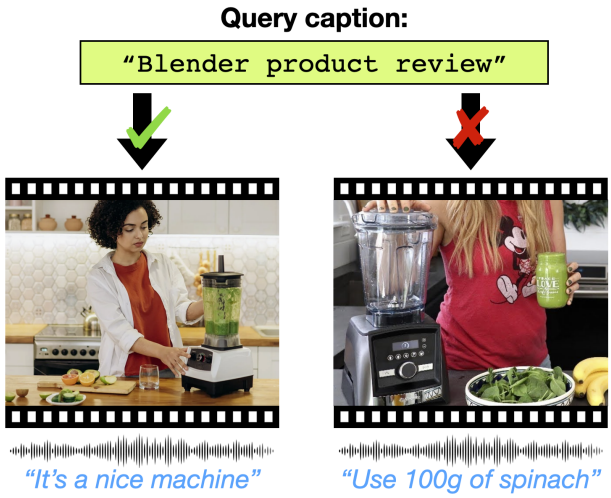

Figure 1. Speech is part of the story! Video retrieval methods that focus on visual inputs alone are likely to miss out on key information (e.g., while both the examples above contain a blender, the speech (in blue) helps identify the one for a product review). In this work, we focus on learning a video encoder to effectively process RGB and audio features, as well as transcribed speech from instructional videos online, through a novel modality masking method. Our approach learns from unlabelled videos, without the need for expensive manual captions.

* Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France.

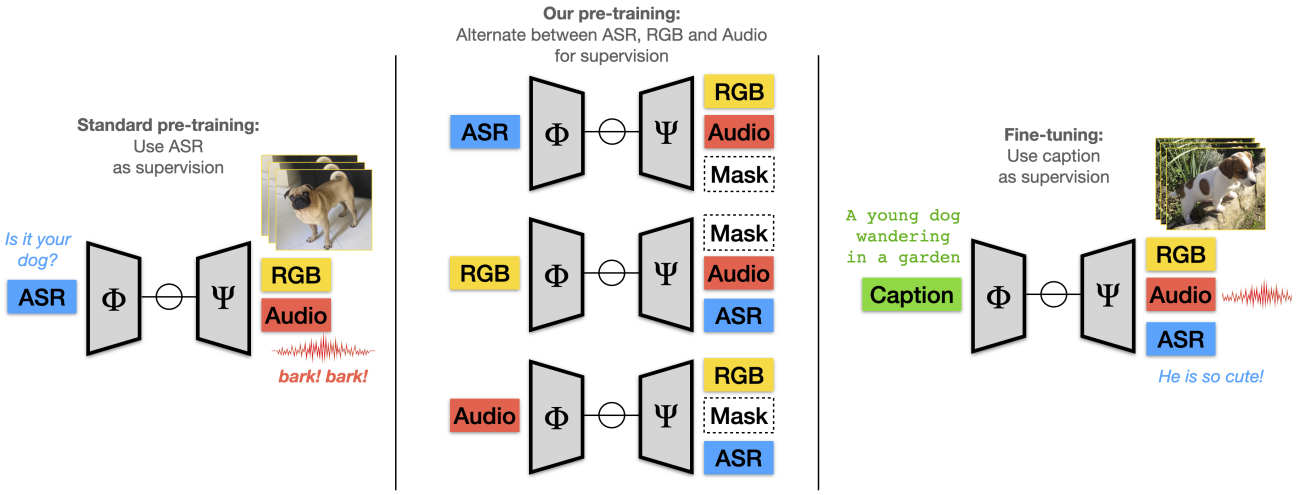

Figure 2. A common paradigm in learning from instructional videos is use transcribed speech (from ASR) (projected here using an encoder Φ) to supervise a video encoder Ψ (left). Instead, we train our video encoder Ψ with three inputs – RGB, audio and transcribed speech (ASR), and alternate between masking and predicting an entire modality at a time (middle). At the time of fine-tuning (right), our video encoder has been pre-trained to use all video modalities.

In this work, we propose a novel pre-training strategy for learning multi-modal fusion from instructional videos (Fig. 2, middle). We learn two encoders - the first being a video encoder (Ψ) that fuses experts from three modalities - RGB, transcribed speech (which we henceforth refer to as ASR for brevity), and audio. During pre-training we use a modality masking strategy, where we mask out an entire modality in the input of the video encoder, and try to predict an encoded version of this modality (encoded using a second encoder (Φ)) from the other modalities. In this manner, the modality being predicted is effectively being used as 'supervision' for the other two. At each batch, we mask a different modality, thereby learning a video encoder that is able to process all the modalities available in the video signal.

We make the following two contributions: (i) We introduce a new pre-training approach for learning video representations that does not require costly manual annotations. Unlike previous works [9, 19, 22, 24], we train our video encoder with three inputs – RGB, audio and ASR, and alternate at each batch which one is used for supervision. At the time of fine-tuning, our video encoder has been pre-trained to use all video modalities. (ii) We obtain competitive results on several standard text-to-video benchmarks.

2. Related work

2.1. The challenging multi-modality of videos

Despite videos being multi-modal, many popular works [4, 15, 18, 22, 27, 36] consider videos as a sequence of visual frames and discard the audio signal. They do not take advantage of the rich and varied additional information present in the audio track, including concepts mentioned in speech and other background sounds, which may be highly relevant to the video semantics. This choice can be explained by the difficulty of jointly processing multiple modalities and the computational cost associated with processing such a high dimensional signal. Accommodating for those challenges, many video benchmarks focus on the visual modality only and ignore the video sounds or speech: The video classification task is usually evaluated on action recognition datasets [17, 32] where the audio is mostly irrelevant to the task. Similarly, the annotations of many video-language datasets are biased toward the visual modality. For example, the LSMDC dataset [30] is annotated with purely visual descriptions, the ActivityNet dataset [16] annotators were required to ignore the audio and the MSVD dataset [6] videos simply do not have audio. In this work, we aim at pre-training a video encoder on all video modalities, including audio and speech. We therefore also evaluate our approach on datasets [3, 19, 40] that do not artificially ignore those modalities.

2.2. Experts for Video-to-Text Retrieval

Because of the small scale of manually annotated text-to-video retrieval datasets as well as the high computational cost of processing pixels and raw audio signal directly, a popular approach for video retrieval has been to use pre-extracted features from 'expert' models. These models are trained for diverse tasks and on multiple modalities such as face [26], scene [39] and object recognition, action classification [5] and sound classification [10]. MoEE [23], CE [20] and MMT [9] all follow this paradigm, with the overall similarity for a video-text pair obtained as a weighted sum of each expert's similarity with the text.

In [11], the authors propose a two-branches architecture that models the interactions between different levels of granularity in both the visual modality and the text modality. More recently, several works [8,28] have shown the superiority of using the CLIP [29] model to extract appearance features, therefore leveraging the 400 million (image, caption) pairs it was trained on.

2.3. Pre-training for Video and Language

Since the release of the HowTo100M dataset [24], a large instructional video dataset, there has been a renewed interest in leveraging large-scale pre-training to improve video-text representations for tasks such as video question-answering [19, 31], text-to-video retrieval [9, 24, 27], action recognition [2, 22, 25, 33] and video captioning [13, 41]. In NLP, BERT [7] and its variants have popularized the 'masked language modelling' self-supervised technique for pre-training: wherein words in the input are randomly masked and the training objective is tasked with predicting their encodings. This technique has been extended to train visual and language encoders (eg. VideoBERT [34], CBT [33], ViLBERT [21], Hero [19] etc). All such works, only mask a proportion of the input (usually 15%). Contrary to natural language, the visual signal and audio signal of a video are continuous and highly redundant. A masked video or audio segment can be easily estimated from its neighboring frames. To address this problem, we mask out an entire modality in the input, forcing our model to learn difficult cross-modal interactions.

3. Methodology

In this section, we first describe the common pre-training approach for learning from instructional videos, where the ASR is used to supervise a visual encoder. We then present our strategy to pre-train a video encoder Ψ on three video modalities: RGB, Audio and ASR, by using each of them to supervise the others in an alternating manner. After pre-training, our video encoder has learnt to attend across all modalities in a video, and can be fine-tuned on video-text datasets for the task of video retrieval.

3.1. Standard Pre-Training

As a video representation learning pre-training strategy, several previous works [9, 22, 24] use the speech modality as supervision to train a video encoder on the other video modalities. Illustrated on the left side of Fig. 2, this approach involves the estimation of a speech representation by a query encoder Φ and a video representation by a video encoder Ψ. The training objective is usually a standard metric learning objective (maximising the similarity between the speech representation and the video representation if they are extracted from the same video, minimizing the similarity between randomly selected speech and video). At the time of fine-tuning (right side of Fig. 2, the query encoder Φ is used to encode the caption while the video encoder Ψ is processing all video modalities, including speech.

The main drawback of this approach is that the video encoder is not pre-trained on speech since that modality is used as pre-training supervision. At the end of pre-training, the video encoder has hence been denied the opportunity to learn complex cross-modal interactions between RGB and speech. The video encoder only learns to process speech during fine-tuning. This is a major limitation as speech may be an integral part of the video signal and encode crucial information for video retrieval.

3.2. Alternating Modality-Masking Pre-Training

We propose a new approach for pre-training a video encoder on a large-scale dataset of raw videos like HowTo100M [24], which does not contain captioning labels. In order for our video encoder to be pre-trained on all video modalities, **including ASR**, we propose to not only use ASR supervision, but to alternate between three objectives (Middle section of Fig.2):

1. Use ASR as supervision to train the video encoder Ψ on processing RGB + Audio as inputs
2. Use RGB as supervision to train the video encoder Ψ on processing Audio + ASR as inputs
3. Use Audio as supervision to train the video encoder Ψ on processing RGB + ASR as inputs

At each training batch, we randomly pick one of those objectives. We therefore randomly pick a modality in $\{RGB, Audio, ASR\}$ to serve as the supervising modality processed by the query encoder, while the other two modalities act as the collaborating modalities processed by the video encoder. Let us take the example of a training batch for which RGB has been selected as the supervising modality. For each video of the batch, its sequence of RGB features will be processed by the query encoder Φ to obtain a query representation. The features of the other two modalities of the video, in this case Audio and ASR, will be processed by the video encoder Ψ to extract both cross-modal and temporal information and obtain a video representation. We will then proceed to optimize the parameters of our encoders so that the query and video representations of a same video are similar while the representations of different videos in the batch are dissimilar.

More formally, for each video clip v_i in the training batch, we separate the expert features in two sets: q_i are the features obtained from the supervising modality and c_i are the features obtained from the collaborating modalities. We then use our query encoder Φ to compute a representation $\Phi(q_i)$ of the supervising modality. Similarly, our video

encoder Ψ will compute a representation $\Psi(c_i)$ of the collaborating modalities.

During fine-tuning (right side of Fig.2), our video encoder Ψ is provided with all the modalities present in the video signal, all of which it has seen before during pre-training, and has hence acquired the ability to model cross-modal complex correlations.

Although our video encoder Ψ only ever receives two modalities at a time during pre-training, they are different at each batch. We therefore need a video encoder capable of processing the three video modalities, but at each batch, one of them (the supervising modality) is "masked out", it is simply not provided to Ψ. In the next section we describe the architecture of that encoder.

3.3. The Multi-Modal Transformer

For our video encoder Ψ, we use the Multi-Modal Transformer described in [9]. It consists in a Transformer encoder that is fed features from different video modalities. The self-attention mechanism of the Transformer allows each token to attend to all the others, therefore being able to process information across both time and modalities. The choice of the MMT architecture for our modality-masking pre-training approach is justified by its capacity to elegantly handle missing modalities. In fact, all the transformer layers parameters are shared across all input features, and therefore modalities. That means that even if one modality is masked from the input of MMT, the parameters of all layers will still be optimized. All parameters needed for the downstream task are optimized at each batch, independently of the chosen objective. This is in contrast to the MoEE style of architecture where there is a dedicated encoding branch for each modality. In the case of a missing expert stream, zeros will be fed, thereby wasting computation for that whole branch.

3.4. Loss Function

For both pre-training and fine-tuning, we optimize both query encoder Φ and video encoder Ψ to provide similar representations when their input features come from the same video clip and dissimilar representations when they come from different clips. We train our model with the bi-directional max-margin ranking loss [14]:

$$\mathcal{L} = \frac{1}{B}\sum_{i=1}^{B}\sum_{j\neq i}\Big[\max(0, s_{ij}-s_{ii}+m)+\max(0, s_{ji}-s_{ii}+m)\Big], \tag{1}$$

where B is the batch size, $s_{ij} = s(\Phi(q_i), \Psi(c_j))$, the similarity score between query representation $\Phi(q_i)$ of video v_i and video representation $\Psi(c_j)$ of video v_j, and m is the margin. This loss enforces the similarity between true representation pairs s_{ii} to be higher than the similarity between negative samples s_{ij} or s_{ji}, for all $i \neq j$, by at least m. This will have the effect of gathering similar captions and videos together in the embedding space, thereby allowing video retrieval to be performed by ranking videos according to their proximity with the query.

3.5. Selection of Modalities

We choose the modalities RGB, Audio and ASR in this work, largely because they often represent complementary aspect of the video signal. Although audio and speech are both extracted from the audio signal, the expert models extracting features for those modalities have been pre-trained on different tasks and present specialized architectures dedicated to those tasks. The CNN used to encode the audio (sounds) features is not capable to extract spoken language nor has it been trained on that task.

4. Experiments

We first describe the text-video datasets that our model is trained and evaluated on (sec. 4.1), then present implementation details (sec. 4.2) and ablation studies (sec. 4.3). Finally, we compare to the state-of-the-art for video retrieval (sec. 4.4).

4.1. Datasets and Metrics

Since the focus of our work is the effective encoding of ASR, audio and visual information, we evaluate on video datasets that contain multimodal captions (i.e., captions that refer to the content in the speech as well). For each dataset, we manually inspect 100 caption-video pairs at random to determine the percentage of captions that are related to what is being said in the video. For example, the caption "Someone talking about love" requires knowledge of the speech in the video, whereas "A woman with a red dress" does not. Results are reported below.

HowTo100M [24] is a very large-scale dataset of over 1M YouTube instructional videos that amounts to about 15 years of video. This dataset was not manually annotated with captions, but is a valuable source of data for self-supervised learning because of the high correlation between visual, audio and speech information in its videos. We only use this dataset to pre-train our model.

How2R [19] features 47,369 clips extracted from the HowTo100M dataset videos and split into a training, validating and testing set. The clips are 17s long on average, and annotated with a caption. Our manual inspection of 100 captions yields 54 captions related to speech. Because it has the same domain as our pre-training dataset, we run ablation studies on this dataset.

MSR-VTT [36] contains 10K YouTube videos with 200K descriptions. Following other works [20], we train on 9K train+val videos and report results on the 1K-A test set. After manual inspection, we find that 12% of the captions are related to speech.

Condensed Movies Dataset (CMD) [3] consists of 33,976 clips extracted from 3,605 movies. Our manual verification process indicates that approximately 60% of CMD descriptions are related to speech.

YouCook2 [40] consists of 176 hours of cooking videos. The videos are segmented into 13,829 clips, each annotated with a sentence describing a step of the recipe. We follow [24] and evaluate our model on 3,350 clips that are not present in HowTo100M. We found about 70% of YouCook2 captions are related to the speech in the video. For example, in the case of a video annotated with "add corn starch", paying attention to the speech: "I now use corn starch" is a strong cue indicating that we are not dealing with flour.

ActivityNet captions [16] consists of 20K YouTube videos annotated with several sentences. We follow [38] and concatenate the sentences to obtain a paragraph annotation for each video. We found that only 1% of the captions relate to speech. The authors of this dataset informed us that the annotators were explicitly asked to ignore the audio. It was turned off by default for the annotation process.

Metrics. We evaluate the performance of our approach on the following standard retrieval metrics: recall at rank N (R@N, higher is better), median rank (MdR, lower is better) and mean rank (MnR, lower is better). For each metric we run the experiment with 3 random seeds and report the mean and standard deviation. We report the test-set performances of our model for the epoch where the validation-set geometric mean of R@1, R@5 and R@10 is maximal.

4.2. Implementation Details

Pre-trained experts. Both encoders use pre-trained expert models for extracting features from each video modality. We use the following 3 experts:

RGB features are extracted from S3D [35] trained on the Kinetics action recognition dataset. We extract one RGB feature of dimension 1024 per second of video.

Audio features are extracted using VGGish model [12] trained on the YouTube8M dataset [1]. We extract one audio feature of dimension 128 per second of video.

ASR transcripts are obtained from the closed captions accompanying videos on YouTube. Words are encoded with BERT-base-cased [7]. We obtain one speech feature of dimension 768 for each wordpiece from the ASR.

Query encoder Φ for pre-training. In the case when the masked modality is either RGB or audio, we encode it using the Multi-Modal-Transformer (MMT) [9] model. We follow [9] and use a 4 layer, 4 head version of MMT. For any modality presented to the query encoder, we do not encode input sequence of features with temporal embeddings. We found that this made the pre-training objective trivially easy to solve - we hypothesise that this is because temporal information allows the encoders to align silences in the ASR and audio features (as both are extracted from the same audiotrack). For example, both encoders would be able to determine the presence and absence of speech from the ASR and audio modalities. The similarity can then be maximised based on this temporal alignment, instead of on the video semantics, leading to performance drops. In the case when the masked modality is ASR, we follow [9] and process the speech words with a pre-trained BERT model. For memory constraints, we limit BERT input to 30 consecutive wordpieces, randomly sampled from the ASR. The representation extracted from the BERT [CLS] token is projected by 3 different gated embedding units (one for each modality) to obtain our query representation $\Phi(q_i)$.

Query encoder Φ for fine-tuning. During fine-tuning, we use the captions as supervision. For fine-tuning on MSRVTT, YouCook2 and ActivityNet, we follow the procedure introduced in MMT [9]: we process the caption with the Bert-based query encoder that we pre-trained earlier. We limit BERT input to 30 consecutive wordpieces, randomly sampled in the caption. On the How2R and CMD datasets, we found out that using the pre-trained Bert-based query encoder for encoding the captions resulted in rapid over-fitting – on the other hand, freezing the weights in the query encoder lead to poor performance. We therefore follow the approach outlined in MoEE [23] and use a net-VLAD layer to aggregate the caption word embeddings (obtained by a frozen BERT model), to obtain the final caption representation. The caption representation is then projected by 3 different gated embedding units.

Video encoder Ψ. This is implemented using the Multi-Modal Transformer(MMT) [9] as our video encoder. We use a 4 layers x 4 heads version of MMT with a dropout probability of 10%, a hidden size d_{model} of 512, and an intermediate size of 3072. We initialize the aggregated embeddings of MMT with a max-pooling aggregation of the modality features. In the case of a masked modality (pre-training) or when a modality is not available in the video, no features are provided to MMT and the aggregated embedding for that modality becomes a zero vector. For all our experiments, we only use the sequences of features extracted by our RGB, audio and ASR pre-trained experts. The parameters of those feature extractors are kept frozen. For memory constraints, we provide the video encoder with sequences of maximum 30 features for the RGB and audio modalities, and maximum 128 features for the ASR. In case more features are available in the video, they are randomly sampled.

Hyperparameters. For each dataset, we estimate the hyperparameters by running a grid search on the corresponding validation set. We use the Adam optimizer for all the experiments.

For pre-training on HowTo100M, we use a batch size of 1,200 videos, an initial learning rate of 1e-4, which we decay by a 0.98 multiplicative factor every 2K optimisation steps, and train for 400K steps. We randomly crop HowTo100M videos into segments of 30 seconds.

For training from scratch or fine-tuning on MSR-VTT or YouCook2, we use a batch size of 32 videos, an initial learning rate of 5e-5, which we decay by a 0.95 multiplicative factor every 1K optimisation steps, and train for 50K steps. For training from scratch or fine-tuning on CMD or How2R, we use an initial learning rate of 5e-5, which we decay by a 0.90 multiplicative factor every 375 optimisation steps, and train for 20K steps. We use a batch size of 32 videos on How2R and 64 videos on CMD. For training from scratch or fine-tuning on ActivityNet, we use a batch size of 24 videos, an initial learning rate of 5e-5, which we decay by a 0.90 multiplicative factor every 1K optimisation steps, and train for 50K steps. For training on HowTo100M, MSR-VTT, YouCook2 or ActivityNet, the bidirectional max-margin ranking loss margin is set to 0.05. For training on How2R or CMD, it is set to 0.2.

Running time. Pre-training our model on HowTo100M takes 12 days on 8 V-100 GPUs. Fine-tuning on MSRVTT, How2R, CMD, YouCook2 or ActivityNet takes about 4 hours on a single V-100 GPU.

4.3. Ablation Analysis

We perform three ablation studies to: (i) show the effect of varying the masking probability of ASR, p, during pre-training; (ii) demonstrate the need of complete modality masking over partial modality masking; and (iii) compare multi-modal retrieval results to those with a single modality.

Effect of the ASR masking probability p. Table 1 shows the impact of different masking probabilities during pre-training on HowTo100M. The probability p refers to the probability of masking our ASR and feeding in only RGB and audio to the candidate encoder (this is the common pre-training paradigm, where ASR is effectively 'supervising' our video encoder). The rest of the time is equally split between masking out audio and RGB. Hence if $p = 0.8$, we mask out ASR 80% of the time, RGB 10% of the time, and audio 10% of the time. Note that this is equivalent to weighting the loss (Eq. 1) differently depending on which modality is masked. We report results on the validation set of How2R after fine-tuning on the training set of How2R. We show that the common pre-training paradigm of always using the ASR to supervise RGB and audio ($p = 1.0$, first line) does not provide the best results. It is better to also use audio and RGB as supervision in order to pre-train the video encoder on speech. For the rest of the experiments, we set $p = 0.8$ during pre-training.

Table 1. The effect of the masking probability for transcribed speech (ASR) p, where $p = 1.00$ refers to the case where ASR is masked 100% of the time, and predicted from audio and RGB. Results are reported on the validation set of How2R after fine-tuning. We note that performance improves when $p < 1$, but remains relatively robust to different values.

	Text $\Longrightarrow$ Video				
p	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
1.00	$3.1_{\pm 0.1}$	$9.9_{\pm 0.3}$	$15.5_{\pm 0.3}$	$97.0_{\pm 2.2}$	$292.3_{\pm 4.7}$
0.90	$2.9_{\pm 0.0}$	$9.5_{\pm 0.0}$	$15.2_{\pm 0.0}$	$96.5_{\pm 0.0}$	$271.9_{\pm 0.0}$
0.80	$\mathbf{3.7}_{\pm 0.1}$	$\mathbf{11.5}_{\pm 0.1}$	$\mathbf{17.8}_{\pm 0.3}$	$\mathbf{79.0}_{\pm 0.0}$	$270.7_{\pm 2.5}$
0.70	$3.5_{\pm 0.1}$	$\mathbf{11.5}_{\pm 0.2}$	$17.6_{\pm 0.1}$	$80.7_{\pm 0.9}$	$\mathbf{267.8}_{\pm 1.4}$
0.33	$3.5_{\pm 0.2}$	$\mathbf{11.5}_{\pm 0.3}$	$\mathbf{17.8}_{\pm 0.2}$	$82.0_{\pm 1.4}$	$269.8_{\pm 1.5}$

Advantage of complete modality masking over partial masking. Several recent works [19, 21, 33, 34] pre-train a video encoder by partially masking a modality (eg: masking 15% of video frames). We instead mask 100% of the modality. We have compared our approach with masking 15%, 50% or 85% of the supervising modality tokens. To not make the task trivial we did not provide the query encoder Φ with 100% of the supervising modality tokens, but only with the feature tokens that were masked from the video encoder Ψ. We used the setting which obtained best results for 100% masking, i.e., ASR is used as supervision for 80% of the batches, audio 10% and RGB 10%.

Recall@10 results on the validation set of How2R are: From scratch (no pre-training): 12.9, Masking 15%: 16.1, Masking 50%: 16.2, Masking 85%: 16.8, Masking 100%: 17.8.

The results for the partial masking pre-training show a lower performance compared to 100% masking. We also noticed that during pre-training the loss for the partial masking experiments was lower than the loss for the 100% masking experiment. This can be attributed to the fact that the query encoder Φ and video encoder Ψ are both provided with some of the supervising modality features, making the pre-training task easier, and therefore less effective. This is particularly the case for the audio and visual modality because of their continuity and high redundancy.

Impact of pre-training on single-modality retrieval. We further evaluate our pre-training approach by fine-tuning the model on a single video modality. In this case, for each video in the How2R validation set, our video encoder is only provided with the features of one modality, either RGB, Audio or ASR. We report the results using R@10 in Fig. 4. When only pre-training with ASR supervision ($p = 1.0$, orange), our video encoder only processes RGB and audio inputs. In this case, we note that pre-training helps when fine-tuning only on the RGB modality, but not on the audio modality. As expected, this setting leads to a performance drop on ASR, as the video encoder has never

seen ASR inputs during pre-training. This is not the case for our alternating modality masking approach where the video encoder was sometimes provided with ASR features and therefore learns to process that modality. We note that our alternating masking approach ($p = 0.8$, green) provides improvements overall, as well as for each modality independently (other than for audio which does not seem to benefit from pre-training).

4.4. Comparison to the State of the Art

Results on How2R are provided in Table 2. The original paper introducing the How2R dataset [19] tackles the task of moment localization in a video clip. We re-purpose the How2R dataset for the task of video retrieval where each moment and its description are considered as a different video-caption pair. We reproduce the MoEE approach [23] on this dataset, and show that our method trained from scratch significantly outperforms MoEE. We also implement the MMT pre-training approach [9] (equivalent to p=1.0) with our features, and compare it with our modality masking pre-training (p=0.8) approach. The large performance improvement obtained with our approach demonstrates the advantage of pre-training the video encoder on speech before fine-tuning on the How2R dataset, which has more than half of its captions related to speech.

Table 2. Text to Video retrieval results on the How2R [19] benchmark. † Our implementation on this dataset using only our RGB, audio and ASR features. sc: trained from scratch on How2R. pt: pre-trained on the HowTo100M dataset, then fine-tuned on How2R.

	Text $\Longrightarrow$ Video				
Method	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
Random	0.0	0.1	0.2	2009.5	2009.5
MoEE (sc) [23]†	$2.2_{\pm 0.1}$	$7.8_{\pm 0.1}$	$12.9_{\pm 0.3}$	$118.7_{\pm 1.2}$	$389.5_{\pm 1.8}$
Ours (sc)	$2.3_{\pm 0.2}$	$8.3_{\pm 0.3}$	$13.6_{\pm 0.2}$	$106.0_{\pm 2.2}$	$312.5_{\pm 1.2}$
MMT (pt) [9]†	$2.9_{\pm 0.0}$	$9.1_{\pm 0.2}$	$14.5_{\pm 0.2}$	$96.0_{\pm 2.2}$	$314.3_{\pm 1.7}$
Ours (pt p=0.8)	$\mathbf{3.4}_{\pm 0.2}$	$\mathbf{11.6}_{\pm 0.2}$	$\mathbf{18.2}_{\pm 0.3}$	$\mathbf{75.3}_{\pm 0.9}$	$\mathbf{277.1}_{\pm 2.3}$

Results on CMD are provided in Table 3. Unlike the original CMD paper [3], we remove actor names from the captions. We re-implement MoEE [23] on this modified dataset using our features, and demonstrate that our pre-training approach provides a significant improvement in performance. Note that this is despite the large variation in domain between pre-training and fine-tuning – while we pre-train on instructional videos from YouTube, CMD consists of short clips extracted from movies.

Table 4 presents results on YouCook2. Due to the high importance of the speech modality in this dataset, pre-training with our approach (pt p=0.8) yields considerable performance improvement, compared to the standard pre-training approach (MMT) that does not pre-train the video encoder on the speech modality.

In Table 5, we compare MSR-VTT results in two different settings: Training from scratch on MSR-VTT (sc) or pre-training on HowTo100M then fine-tuning on MSR-VTT (pt). When training from scratch, our method has a small drop in performance, when compared to MMT [9]. This is likely due to our approach using only 3 modalities instead of 7. Our method's performance is also weaker than a recent approach SSB [27] that uses a modified version of MMT. In the HowTo100M pre-training setting however, our modality masking approach outperforms the standard pre-training used in MMT, even if only 12% of MSR-VTT annotations are related to speech. Our results are competitive wrt SSB. We also show qualitative results of our method on this dataset in Fig. 3. Note how we perform well in the examples shown in the top two rows – both the queries refer to the contents of speech. In the second row, while the correct video is retrieved at rank 5, the other videos in the top 5 also describe school systems, demonstrating the difficulty of the dataset where often a caption may be equally relevant to a number of videos.

Table 3. Results on the Condensed Movies Dataset (CMD) [3]. † Our implementation on this dataset using only our RGB, audio and ASR features. ‡ Our implementation on this dataset using the code and all the features provided by the authors of CMD [4]. sc: trained from scratch on CMD. pt: pre-trained on the HowTo100M dataset, then fine-tuned on CMD.

	Text $\Longrightarrow$ Video				
Method	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
Random	0.0	0.1	0.2	3284.5	3284.5
MoEE (sc) [23]†	$3.2_{\pm 0.1}$	$9.9_{\pm 0.3}$	$14.9_{\pm 0.4}$	$142.7_{\pm 0.5}$	$532.7_{\pm 5.7}$
CMD (sc) [3]‡	2.6	10.2	16.2	102	377.7
Ours (sc)	$4.6_{\pm 0.1}$	$13.5_{\pm 0.2}$	$19.5_{\pm 0.1}$	$89.7_{\pm 1.2}$	$396.5_{\pm 5.5}$
Ours (pt p=0.8)	$\mathbf{5.8}_{\pm 0.2}$	$\mathbf{15.8}_{\pm 0.2}$	$\mathbf{22.4}_{\pm 0.1}$	$\mathbf{73.7}_{\pm 1.7}$	$\mathbf{369.6}_{\pm 4.6}$

Table 4. Results on the YouCook2 dataset [40]. † Our implementation on this dataset. sc: trained from scratch on YouCook2. pt: pretrained on the HowTo100M dataset, then fine-tuned on YouCook2.

	Text $\Longrightarrow$ Video				
Method	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
Random	0.03	0.15	0.3	1675	1675
Ours (sc)	$16.6_{\pm 0.2}$	$37.4_{\pm 0.3}$	$48.3_{\pm 0.1}$	$12.0_{\pm 0.0}$	$95.5_{\pm 3.4}$
HT (pt) [24]	8.2	24.5	35.3	24	-
COOT (sc) [11]	$16.7_{\pm 0.4}$	$40.2_{\pm 0.3}$	$52.3_{\pm 0.5}$	$9.0_{\pm 0.0}$	-
MMT (pt) [9]†	$17.2_{\pm 0.4}$	$39.5_{\pm 0.7}$	$51.0_{\pm 0.5}$	$10.0_{\pm 0.0}$	$68.2_{\pm 0.9}$
Ours (pt p=0.8)	$\mathbf{23.2}_{\pm 0.5}$	$\mathbf{48.0}_{\pm 0.7}$	$\mathbf{58.6}_{\pm 0.8}$	$\mathbf{6.0}_{\pm 0.0}$	$\mathbf{60.4}_{\pm 3.0}$

Results on ActivityNet are presented in Table 6. The annotators of this dataset were explicitly required to ignore the audio track when describing the videos, therefore focusing the descriptions towards the visual modality. Our multimodal pre-training approach hence yields similar results to the previous state-of-the-art method (SSB [27]).

5. Conclusion

We present a new pre-training method for learning a multimodal video encoder. It consists of an alternating modality masking strategy, where we mask and predict a different modality at each batch using the other available

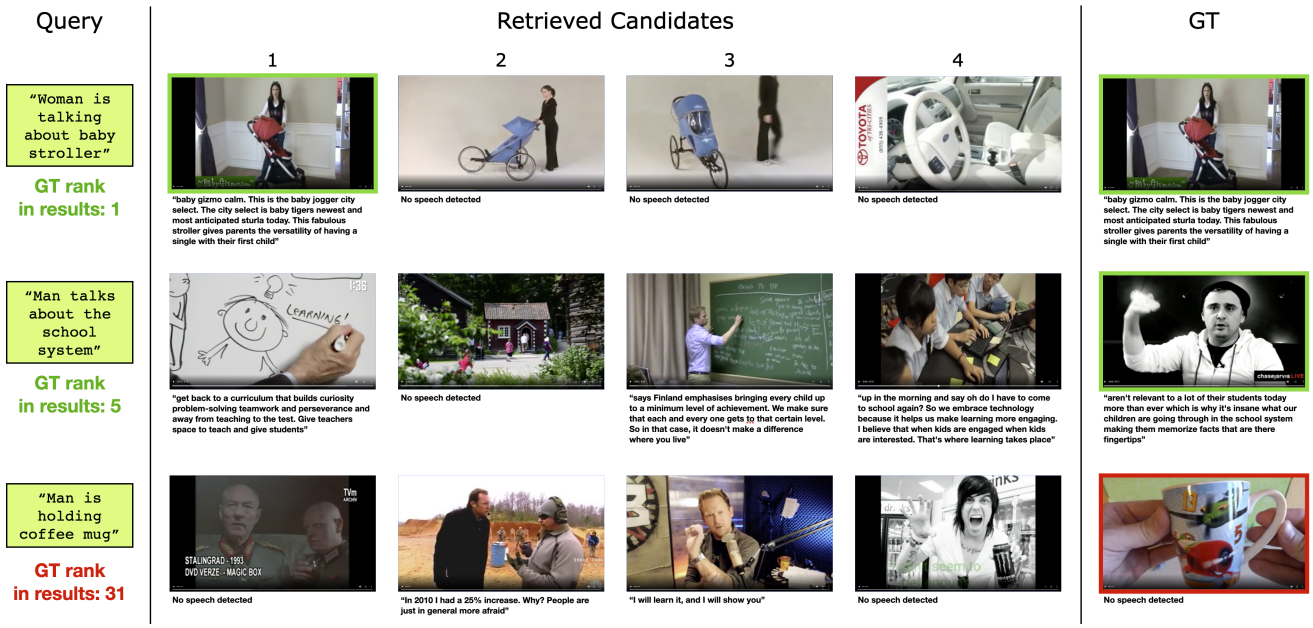

Figure 3. Qualitative results of our retrieval method on the MSR-VTT dataset. For each query, we show frames and ASR from the top 4 ranked videos as well as for the ground truth video. We indicate the rank of the ground-truth video in our retrieval results (highlighted in green when it is in the top-5 retrieved results, or red otherwise) on the left under the query. Note that there are 1000 candidate videos in the test set. (Best viewed on screen.)

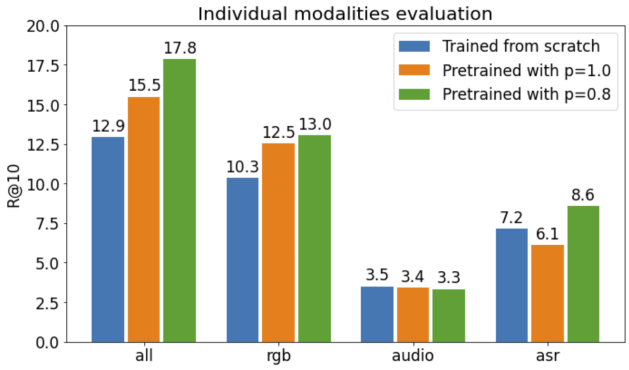

Figure 4. Impact of the pre-training approach on the retrieval of a single modality. We report results on the val set of How2R using R@10. (Best viewed in colour.)

modalities. We show that this allows us to effectively pre-train a video encoder to jointly process RGB, audio and ASR, even on unlabelled datasets without manually-generated captions. Our method produces competitive results on five downstream video retrieval benchmarks. It is particularly suitable when user queries relate to the spoken language in the videos.

Acknowledgements. This work was supported in part by the ANR grant AVENUE (ANR-18-CE23-0011).

Table 5. Comparison to state of the art on the 1K-A split [20] of the MSR-VTT dataset [36]. sc: trained from scratch on MSR-VTT. pt: pre-trained on the HowTo100M dataset, then fine-tuned on MSR-VTT.

	Text $\Longrightarrow$ Video				
Method	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
Random	0.1	0.5	1.0	500.5	500.5
JSFusion (sc) [37]	10.2	31.2	43.2	13	-
HT (sc) [24]	12.1	35.0	48.0	12	-
CE (sc) [20]	$20.9_{\pm 1.2}$	$48.8_{\pm 0.6}$	$62.4_{\pm 0.8}$	$6.0_{\pm 0.0}$	$28.2_{\pm 0.8}$
MMT (sc) [9]	$24.6_{\pm 0.4}$	$54.0_{\pm 0.2}$	$67.1_{\pm 0.5}$	$4.0_{\pm 0.0}$	$26.7_{\pm 0.9}$
Ours (sc)	$22.5_{\pm 0.9}$	$53.2_{\pm 1.5}$	$67.1_{\pm 0.4}$	$4.7_{\pm 0.5}$	$25.8_{\pm 0.3}$
SSB (sc) [27]	27.4	56.3	67.7	3.0	-
HT (pt) [24]	14.9	40.2	52.8	9	-
Hero (pt) [19]	20.5	47.6	60.9	-	-
FiT (pt) [4]	24.1	-	63.9	5	-
MMT (pt) [9]	$26.6_{\pm 1.0}$	$57.1_{\pm 1.0}$	$69.6_{\pm 0.0}$	$24.0_{\pm 0.8}$	
SSB (pt) [27]	**30.1**	58.5	69.3	**3.0**	-
Ours (pt p=0.8)	$28.7_{\pm 0.7}$	**59.5**$_{\pm 0.7}$	**70.3**$_{\pm 0.7}$	$3.8_{\pm 0.2}$	**23.0**$_{\pm 0.5}$

Table 6. Paragraph to video retrieval performance on the ActivityNet dataset [16]. sc: trained from scratch on ActivityNet. pt: pre-trained on the HowTo100M dataset, then fine-tuned on ActivityNet.

	Text $\Longrightarrow$ Video				
Method	R@1↑	R@5↑	R@50↑	MdR↓	MnR↓
Random	0.02	0.1	1.02	2458.5	2458.5
FSE (sc) [38]	$18.2_{\pm 0.2}$	$44.8_{\pm 0.4}$	$89.1_{\pm 0.3}$	7	-
CE (sc) [20]	$18.2_{\pm 0.3}$	$47.7_{\pm 0.4}$	$6.0_{\pm 0.0}$	$23.1_{\pm 0.5}$	
HSE (sc) [38]	20.5	49.3	-	-	-
MMT (pt) [9]	$28.7_{\pm 0.2}$	$61.4_{\pm 0.2}$	$94.5_{\pm 0.0}$	$3.3_{\pm 0.5}$	**16.0**$_{\pm 0.4}$
SSB (pt) [27]	**29.2**	61.6	**94.7**	**3.0**	-
Ours (pt p=0.8)	$29.0_{\pm 0.5}$	**61.7**$_{\pm 0.3}$	$94.6_{\pm 0.2}$	$4.0_{\pm 0.0}$	$16.8_{\pm 0.5}$

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Apostol (Paul) Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. YouTube-8M: A large-scale video classification benchmark. In *arXiv:1609.08675*, 2016.

[2] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020.

[3] Max Bain, Arsha Nagrani, Andrew Brown, and Andrew Zisserman. Condensed movies: Story based retrieval with contextual embeddings. In *ACCV*, 2020.

[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] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

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

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

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

[9] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal Transformer for Video Retrieval. In *ECCV*, 2020.

[10] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *ICASSP*, 2017.

[11] Simon Ging, Mohammadreza Zolfaghari, Hamed Pirsiavash, and Thomas Brox. COOT: Cooperative hierarchical transformer for video-text representation learning. In *NeurIPS*, 2020.

[12] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. CNN architectures for large-scale audio classification. In *ICASSP*, 2017.

[13] Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. Multimodal pretraining for dense video captioning. In *AACL*, 2020.

[14] Andrej Karpathy, Armand Joulin, and Li Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *NeurIPS*, 2014.

[15] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, 2017.

[16] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, 2017.

[17] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *ICCV*, 2011.

[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] 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.

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

[21] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019.

[22] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-End Learning of Visual Representations from Uncurated Instructional Videos. In *CVPR*, 2020.

[23] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a text-video embedding from incomplete and heterogeneous data. *ArXiv*, abs/1804.02516, 2018.

[24] 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.

[25] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *CVPR*, 2020.

[26] Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *BMVC*, 2015.

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

[28] Jesús Andrés Portillo-Quintero, José Carlos Ortiz-Bayliss, and Hugo Terashima-Marín. A straightforward framework for video retrieval using clip. In *MCPR*, 2021.

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

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

[31] Paul Hongsuck Seo, Arsha Nagrani, and Cordelia Schmid. Look before you speak: Visually contextualized utterances. In *CVPR*, 2021.

[32] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.

[33] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer. *arXiv 1906.05743*, 2019.

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

[35] 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.

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

[37] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018.

[38] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *ECCV*, 2018.

[39] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Trans. PAMI*, 40(6):1452–1464, 2017.

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

[41] Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *CVPR*, 2018.