UnLoc: A Unified Framework for Video Localization Tasks

Shen Yan[1]* Xuehan Xiong[1]* Arsha Nagrani[1] Anurag Arnab[1] Zhonghao Wang[2]†
Weina Ge[1] David Ross[1] Cordelia Schmid[1]
[1]Google Research [2]University of Illinois at Urbana-Champaign
{shenyan, xxman, anagrani, aarnab, dross, cordelias}@google.com {wzhonghao95}@gmail.com

Abstract

While large-scale image-text pretrained models such as CLIP have been used for multiple video-level tasks on trimmed videos, their use for temporal localization in untrimmed videos is still a relatively unexplored task. We design a new approach for this called UnLoc, which uses pretrained image and text towers, and feeds tokens to a video-text fusion model. The output of the fusion module are then used to construct a feature pyramid in which each level connects to a head to predict a per-frame relevancy score and start/end time displacements. Unlike previous works, our architecture enables Moment Retrieval, Temporal Localization, and Action Segmentation with a single stage model, without the need for action proposals, motion based pretrained features or representation masking. Unlike specialized models, we achieve state of the art results on all three different localization tasks with a unified approach. Code is available at: https://github.com/google-research/scenic.

1. Introduction

Contrastive vision-language pretraining has been shown to learn powerful feature representations, and moreover enables open-set inference on a wide range of tasks [55, 27]. As a result, pretrained models such as CLIP [55] have been adapted to multiple diverse tasks including video classification [52, 37], object detection [46] and segmentation [18].

In this paper, we study how to adapt large-scale, contrastively trained image-text models to untrimmed video understanding tasks that involve localization. While CLIP has been used widely for trimmed video tasks (classification [52, 37] or retrieval [4]), its use on long, untrimmed video is still in a nascent stage. Long videos come with multiple challenges – CLIP is pretrained on images only, and localization in untrimmed videos requires exploiting *fine-grained* temporal structured information in videos. In particular, it is challenging for image and language models to learn properties of temporal backgrounds (with respective to foreground actions) during training. In contrast, natural videos often come with a large, variable proportion of background and detecting specific actions is critical for localization tasks [49]. Finally, localization in long untrimmed videos also typically involves detecting events at multiple temporal scales. Consequently, existing approaches that use CLIP typically focus on a two-stage approach involving off-the-shelf proposal generators [28], or use temporal features such as I3D [48] or C3D [60]. In contrast, we propose an end-to-end trainable one-stage approach starting from a CLIP two tower model only.

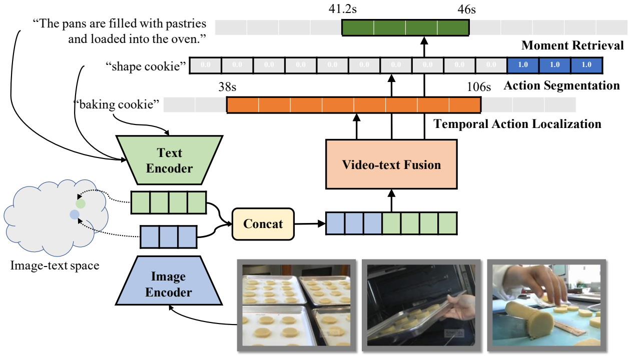

Figure 1. **Applying two-tower CLIP to video localization tasks:** We propose **UnLoc**, a single stage, unified model that achieves state of the art results on 3 different video localization tasks - moment retrieval, temporal action localization and action segmentation. UnLoc leverages a two-tower model (with a vision and text encoder) in conjunction with a video-text fusion module and feature pyramid to perform mid-level feature fusion without the need for any temporal proposals.

We focus specifically on three different video localization tasks - Moment Retrieval (MR) [29, 16], Temporal Action Localization (TAL) [21, 26] and Action Segmentation (AS) [62]. These tasks have typically been studied separately, with different techniques proposed for each task. We show how we can use a single, unified approach, to address

*Equal contribution.
†Work done while an intern at Google

all of these tasks, without using any external proposals. We do this by leveraging a two-tower model (with a vision and text encoder), in conjunction with a single video-text fusion module, which performs mid-level fusion of text and visual tokens (Figure 1). Our two tower model can naturally handle tasks such as moment retrieval which contain both video and text as input modalities, and can be used for open-set inference in other tasks such as temporal action localization and action segmentation. While many works use the visual encoder only [58, 8, 20, 79], we believe that the language priors learnt with the pretrained text encoder can contain useful information and should be leveraged together with the image encoder early in the model design (particularly for open-set evaluation), and not right at the end for similarity computation. Inspired by existing object detection works [31], we also use the output frame tokens from our fusion module to construct a feature pyramid, to enable understanding at multiple temporal scales.

Our approach achieves state-of-the-art results across all three video localization tasks - MR [29, 16], TAL [21, 26] and AS [62]. We also perform thorough ablation studies, studying the effect of modelling choices across a range of tasks.

2. Related Work

Models based on CLIP for localization. Most works use CLIP for video level tasks that operate on short, trimmed clips, for example for classification tasks (*e.g.* ActionCLIP [68] and X-CLIP [52]). EVL [37] also adapts CLIP to video classification, but does so by training a small number of extra parameters. CLIP has also been used for other video level tasks such as text-video retrieval, as done in CLIP4CLIP [43] and CLIP-Hitchikers [4]. A number of works also use CLIP for tasks such as object detection [46, 85] and segmentation [19, 13]. Works that use CLIP for localization tasks in untrimmed videos, on the other hand are less common. Vid2Seq [72] uses CLIP features for dense video captioning, where temporal boundaries and captions are predicted together. Most works for localization however still reply heavily on I3D [7], C3D [64], R(2+1)D [65], VGG [59], or SlowFast [14] features for moment retrieval [60, 76, 81, 80, 50], temporal action localization [49, 87, 79] and action segmentation [42].

Temporal Action Localization (TAL). Supervised learning-based temporal action localization can be summarized into two-stage [57, 8, 83, 32, 34] and single-stage methods [58, 33, 51, 79]. More recently, EffPrompt [28] uses a two-stage sequential localization and classification architecture for zero-shot action localization, with the first stage consisting of action proposal generation with an off-the-shelf pre-trained proposal detector (e.g., BMN [32]), followed by proposal classification using CLIP features. We aim to build a proposal-free framework and directly regress the temporal location of the corresponding class labels or queries by using the fused video-text features. The closest to our method is STALE [49], which trains a single-stage model for zero-shot localization and classification, using representation masking for frame level localization. Unlike STALE, which evaluates on only TAL, we present a single unified method for MR, TAL and AS, and also introduce a feature pyramid for multi-scale reasoning.

Moment Retrieval (MR). Unlike TAL, where class names are predefined used a closed-form vocabulary, MR aims to find the relevant clip in an untrimmed video for a given open-ended natural language query. Early works use sliding windows over video sequences to generate video segment proposals [22, 16], after which the proposals are ranked by their similarity to the query. This ignores the fine-grained relationships between video frames and the words in sentences. Anchor-based methods [9, 67, 75] avoid proposal generation by assigning each frame with multi-scale anchors sequentially and use these to obtain more fine-grained matchings between video and text. Regression-based methods [10, 76, 81, 48, 30, 40] involve learning cross-modal interactions to directly predict the temporal boundaries of a target moment without the need for proposal generation. Our work belongs to this category, unlike works that tend to use the text tower only at the end to compute similarity scores [23, 78, 17, 81, 40], we fuse image and text tokens early on in our model to better leverage language priors from the pretrained CLIP text tower.

Action Segmentation (AS). Action segmentation involves assigning a pre-defined label to each token or frame in a untrimmed long video, which helps to distinguish meaningful video segments from other tokens or frames [62]. While previous works [61, 44, 70, 42] pretrained their models on HowTo100M [45], our approach involves initializing models with pretrained CLIP models. CLIP was trained on pairs of web images and text, which may be less prone to noise compared to ASR and clip pairs.

3. Method

Our model unifies three tasks: MR, TAL and AS, which we first define in Sec. 3.1. As shown in Fig. 2, our model (Sec. 3.2) first tokenizes a (video, text) pair and then fuses information from the two modalities together with a simple video-text fusion module. To capture the multi-scale information needed for localization, we then construct a Feature Pyramid (Sec. 3.3) on the output of the video-text fusion module. These multi-scale features are then fed into a task-specifc Head module (Sec. 3.4) to localize activities or "ground" a language description.

3.1. Tasks

Moment Retrieval (MR), also known as Video Grounding, is the task of matching a given language description

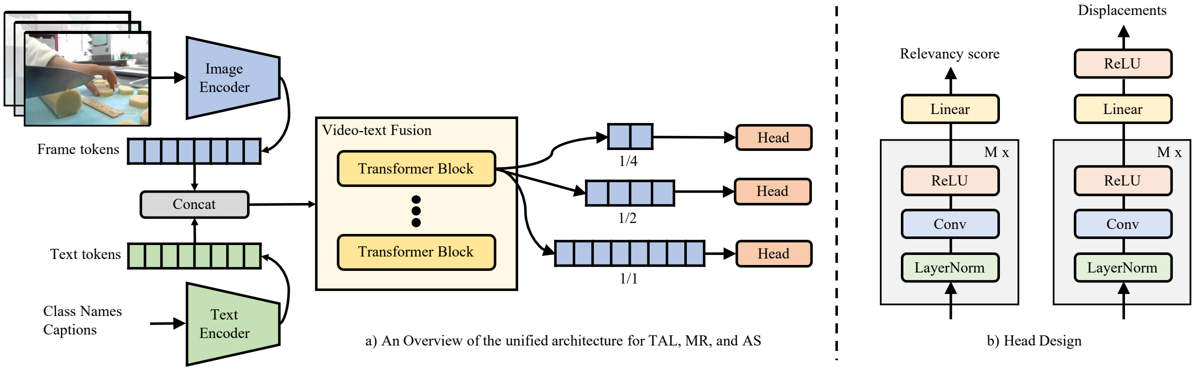

Figure 2. **Overview of our method UnLoc.** Given a video and text (e.g., class names in TAL/AS or captions in MR) pair, first they are tokenized and encoded by a pair of image and text encoders. Frame and text tokens are concatenated into a long sequence and then fed into a transformer for fusion. Frame tokens from the last transformer layer are used to construct a feature pyramid in which each level connects to a head to predict a per-frame relevancy score and start/end time displacements. No text token is used to construct the feature pyramid since text information has already been "fused" into the frame tokens via self-attention. We show a 3 layer feature pyramid for simplicity. All heads across different pyramid levels share the same weights.

(query) to specific video segments in an untrimmed video. *Temporal Action Localization (TAL)* aims to detect events in a video and output the corresponding start- and end-timestamps. One key difference from MR is that events in TAL are from a predefined closed-vocabulary set, often described by a short phrase (e.g., "baking cookies"). Finally, similar to Semantic Segmentation, which parses images into semantic categories at a pixel level, *Action Segmentation (AS)* involves producing activity labels at a frame level. Also, for this task the labels are typically predefined from a closed-vocabulary set.

3.2. A unified architecture

Our model takes (video, text) pairs as inputs, and for each frame in the video it outputs a relevancy score between the frame and the input text, as well as the time differences between the frame and the start/end timestamps of the predicted segment. The target relevancy score is set to 1 if a frame falls within the labeled segment, otherwise zero. In the case of TAL and AS, we use class labels as the input texts while in MR, text queries are used as input texts. For each video we form C (video, text) pairs where C is the number of classes in TAL and AS and for MR C is the number of captions associated with this video.

Fig. 2 gives an overview of our proposed architecture. The input pair is first tokenized and encoded by a pair of image and text encoders. The two encoders are initialized from a pretrained, CLIP visual language model [55]. The two encoders come from the same pretrained model – pretrained by aligning image and text pairs with a contrastive loss, which provides a strong prior on measuring the relevancy between each frame and the input text. This is one of key contributing factors to the success of our model. As Sec. 4.2 will show, using "unpaired" image/text encoders indeed diminishes the performance.

After tokenization and encoding, the input video and text are represented by N frame tokens and T text tokens. We then form a new sequence by concatenating N frame tokens with either a single token (e.g., CLS) representing the whole text sequence or all T text tokens (See Sec. 4.2 for ablation). The concatenated sequence is then fed into a video-text fusion module. In this work, we implement this fusion module using a transformer encoder [66, 12]. This encoder performs two key functions – (i) it is a temporal encoder, able to model inter-frame correspondences omitted by the image-only CLIP model, and (ii) it can also function as a refinement network, with the ability to correct mistakes made by the CLIP model. After fusion, only frame tokens $\mathbf{X}^c \in \mathbb{R}^{N\times K}$ are used to construct a feature pyramid where each level is created by downsampling the original sequence using strided convolutions where c is the index of the class or caption and K denotes the hidden size of the token. This process is repeated for all class labels/captions. Text tokens are omitted from this construction because their information has been incorporated into the frame tokens by the fusion module, and they do not correspond to any timestamps.

Finally, each pyramid level connects to a Head module to predict a per-frame relevancy score, $\hat{\mathbf{y}}_l^c \in \mathbb{R}^{N_l}$, and start/end time displacements, $\hat{\mathbf{t}}_l^c \in \mathbb{R}^{N_l\times 2}$, where N_l denotes the number of features in pyramid level l. The final number of predictions is $\sum_{i=1}^{L}\frac{N}{2^{i-1}}$, and is therefore greater than N if there is more than one level in the feature pyramid. For example, if we construct a 3-level feature pyramid the total number of predictions will be $N + N/2 + N/4$. Each prediction is expanded into a temporal segment by applying the predicted displacements to its frame timestamp. Given these temporal segments for all pyramid layers, we filter out overlapping segments during inference with soft non-maximal suppression (SoftNMS) [5].

3.3. Feature pyramid

A feature pyramid can improve a model's capability to detect events at different scales. For example, features from the top level can detect events with a long duration while bottom-level features can localise short segments. Feature Pyramid Networks (FPN [35]) have been used extensively in object detection for images to pass richer semantic information from a higher level in the CNNs to lower level feature maps that have higher spatial resolution. We propose another simpler structure inspired by ViTDet [31] by removing the lateral and top-down connections in the FPN. Since the last layer in the transformer encoder contains the most semantic information [56] and shares the same temporal resolution as the first one, the lateral and top-down connections are no longer required. The feature pyramid is constructed by applying convolution with different strides to the output tokens from the last transformer layer in the video-text fusion module (See Fig. 2a). Note that text tokens are not used during the feature pyramid construction since their information has been fused into the frame tokens. This simpler design removes the downsampling step in the encoder and allows us to share the same architecture used in pretraining stage (See Sec. 4.1.2 for more details). Similar to findings in [31], our ablation (Sec. 4.2) shows that this simpler design outperforms FPN on TAL as it introduces less additional layers to the pretrained model. AS is a frame-level task so features from only the bottom level in the feature pyramid are used for prediction.

3.4. Head design

As shown in Fig. 2b, we have two heads, one for relevancy score prediction and the other for displacement regression. Although the two heads share the same structure their weights are not shared. Our head design following [79] is simple consisting of M 1D convolution blocks where each block is made of three operations: Layer Normalization [3], 1D convolution, and a ReLU activation [15]. A convolution (e.g., a local operation) is used to encourage nearby frames to share the same label. At the end of each head, a linear layer is learned to predict per-frame relevancy scores $\hat{\mathbf{y}}^c \in \mathbb{R}^{N\times 1}$ or to predict per-frame start/end time displacements $\Delta\hat{\mathbf{t}}^c \in \mathbb{R}^{N\times 2}$:

$$\hat{\mathbf{y}}^c = \mathbf{Z}^c \mathbf{w}_{cls} + b_{cls} \tag{1}$$

$$\Delta\hat{\mathbf{t}}^c = \text{relu}(\mathbf{Z}^c \mathbf{w}_{reg} + \mathbf{b}_{reg}) \tag{2}$$

where $\mathbf{Z}^c$ are the activations of frame tokens $\mathbf{X}^c$ after convolution blocks, $\mathbf{w}_{cls} \in \mathbb{R}^{K\times 1}$ and $b_{cls} \in \mathbb{R}^{1\times 1}$ are the weights and bias for the classification head, and $\mathbf{w}_{reg} \in \mathbb{R}^{K\times 2}$ and $\mathbf{b}_{reg} \in \mathbb{R}^{1\times 2}$ are the weights and biases for the regression head. We limit the predicted displacements to be greater or equal to zero through a ReLU non-linearity. Eqs. 1 and 2 are repeated to generate scores and displacements for every class/caption and the same learned weight and bias terms are shared. For AS only the relevancy scoring head is used. One key difference from [79] is that our model predicts a different start/end time displacement for each class while [79] predicts one displacement $\Delta\hat{\mathbf{t}} \in \mathbb{R}^{N\times 2}$ shared among all classes, which assumes that there is no overlapping segment in the video.

3.5. Loss function

For AS, we use sigmoid cross entropy loss to measure the relevance between a frame and class label. For TAL and MR, we use the focal loss [36] for the relevancy scoring head as class imbalance is a known issue in one-stage detectors [36]. For the regression head we experiment with four popular regression losses, L1, IoU, DIoU [84], and L1+IoU. The L1 loss computes the absolute distance between the predicted and the ground truth start/end times.The IoU loss directly optimizes the intersection of union objective, which is defined as

$$\begin{aligned} L_{iou} &= 1 - IoU(\Delta\hat{s}, \Delta\hat{e}) \\ &= 1 - \frac{\min(\Delta\hat{s}, \Delta s) + \min(\Delta\hat{e}, \Delta e)}{\max(\Delta\hat{s}, \Delta s) + \max(\Delta\hat{e}, \Delta e)} \end{aligned}$$

where $\Delta\hat{s}$, $\Delta\hat{e}$ and Δs, Δe are the predicted and the ground truth displacements to the start/end times. If $\Delta\hat{s}$ or $\Delta\hat{e}$ is zero, its gradient will also be zero, which could happen due to poor initialization. Distance IoU (DIoU [84]) is proposed to address the zero-gradient issue by also taking into account the distance between the two centers of the ground truth box and the predicted box.We end up using L1 loss based on the ablation in Sec. 4.2 and also apply a weight factor α to balance between the focal loss and L1 loss.

4. Experimental Evaluation

We first describe datasets, evaluation metrics and implementation details in Sec. 4.1. We then provide a number of ablations on our architecture design, use of the text encoder, video-text fusion module and finetuning strategies (Sec. 4.2). Finally, we show the results of our method compared to the state-of-the-art in Sec 4.3.

4.1. Experimental setup

4.1.1 Datasets and evaluation metrics

Moment retrieval. *ActivityNet Captions* [29] contains 20,000 videos and 100,000 segments where each is annotated with a caption by human. On average each caption contains 13.5 words and videos have an average duration of 2 minutes. The dataset is divided into three splits, train, val_1, and val_2. Following [76, 60] we use train split for training, val_1 for validation and val_2 for testing. *Charades-STA* [16] contains 6,672 videos and 16,128 segment/caption pairs, where 12,408 pairs are used for training and 3720 for testing. Each video is annotated with 2.4 segments on average and each has an average duration of 8.2 seconds. *QVHighlights [30]* includes over 10,148 cropped videos (150s long), and each video is annotated with at least one query describing the relevant moments (24.6s in average). In total, there are 10,310 text queries with 18,367 associated moments. Following [30, 40], we use train split for training and val split for testing. The most commonly used metric for moment retrieval is the average recall at k computed under different temporal Intersection over Union (IoU) thresholds, which is defined as the percentage of at least one of the top-k predicted segments having a larger temporal IoU than the threshold with the ground truth segment, i.e. Recall@K, IoU=[0.5, 0.7].

Temporal action localization. *ActivityNet 1.3* [21] is a collection of 20,000 untrimmed videos focusing on human actions. Most videos contain only one labeled segment and segments in one video are from the same action class. The dataset is divided into three subsets, train, validation, and test. Following standard practice [34, 32, 71, 79], we train our models on the training set and report results on the validation set. The standard evaluation metric for temporal localization is mean Average Precision (mAP) computed under different temporal IoU thresholds. We report mAP under an IoU threshold of 0.5, denoted as mAP@0.5IoU. We also report results for the zero-shot setting, following the data split protocols proposed by [28, 49]: 1) training on 50% of the action labels and testing on the remaining 50%; 2) training on 75% of the labels and testing on the rest 25%. These are created using 10 random splits of the data, following [28, 49]. In the rest of paper, we use ANet TAL and ANet MR to denote *ActivityNet 1.3* and *ActivityNet Captions*, respectively.

Action segmentation. The *COIN* [62] dataset consists of 11,827 training videos and 2,797 testing videos. Each video is labeled with an average of 3.9 segments where each segment lasts 14.9 seconds on average. The segment labels describe a step needed to complete a task, such as “take out the old bulb”, “install the new bulb”, etc. Frame accuracy is the primary metric used in the COIN action segmentation task, which is defined as the number of correctly predicted frames divided by the total number of frames. However given how a large proportion of the frames are labelled as background (58.9%), a naive majority-class prediction model will already get an accuracy of 58.9% (shown in the first row of Table 7). Hence we also report mean Average Precision (mAP), which averages AP over the classes (excluding background) and is therefore not directly impacted by the large proportion of background.

4.1.2 Implementation details

Model Architecture: In UnLoc-Base and Large models, the image and text encoders follow the same architecture used in CLIP-B and CLIP-L. The video-text fusion module is implemented using a 6-layer Transformer and the hidden size is set to 512 and 768 for UnLoc-B and UnLoc-L and the MLP dimension is set to 2048 and 3072, respectively. We construct a 4-layer feature pyramid from the last layer in the video-text fusion module following the procedure described in Section 3.3. Following [79], an output regression range is specified for each level in the pyramid, which is set to [0, 4], [4, 8], [8, 16], [16, inf], respectively ordered from bottom to the top. All heads across different pyramid levels share the same weights, and are randomly initialized.
Pretraining: Our models are pretrained on Kinetics (K700 [6] for our best models, K400 for ablations). The pretraining task is a 400/700-way binary classification problem using a sigmoid cross entropy loss. For example, for each video we feed all class names into the text tower and the objective is to classify whether or not the video matches any of the class names. During Kinetics pretraining, the image encoder is finetuned and the text encoder is kept frozen to avoid catastrophic forgetting due to the fact that we are finetuning on a small fixed set of vocabulary in Kinetics. The video-text fusion module is always finetuned.
Training: In training the frames are first resized to have a shorter side of 256 and models are trained on a random crop of size 224 × 224. For TAL and AS class names are augmented using Kinetics prompts released by [55], e.g., “`a video of a person doing {label}`”. Unless specified otherwise, all TAL and MR models are trained on 128 frames evenly spaced sampled across the whole video. This follows the sampling strategy adopted by [79] to deal with videos of varying lengths. Unless specified otherwise, for AS on the COIN dataset, we extract the RGB frames at 2FPS, which is the labelling resolution. We randomly sample 512 consecutive frames and apply padding for videos with less than 512 frames. All models are trained using synchronous SGD with a momentum of 0.9, with a batch size of 64. We follow [2] and apply the same data augmentation and regularization schemes [11, 24], which were used by [63] to train vision transformers more effectively. For more implementation details and hyperparameters, we refer readers to the appendix and code.

Inference During inference, our results are obtained evaluating a single central crop of 224 × 224. For AS on COIN, we run our model in a non-overlapping sliding window fashion with a window size of 512 frames. For TAL and AS, we report two results, one using the first prompt and the other by averaging all 28 context prompts, which is defined as prompt ensembling in [55].

Losses				Feature Pyramid			# conv layers			
L1	IoU	L1+IoU	DIoU	No	FPN	ViTDet	1	2	3	4
54.6	54.0	53.9	54.1	47.3	53.8	**54.7**	52.5	53.4	**54.7**	54.5

Table 1. **Effect of architecture design and losses.** Results are presented on the ANet TAL for mAP@0.5IoU. We compare 4 popular regression losses, two types of feature pyramids (and no pyramid), and the number of convolutional layers in the localization heads.

4.2. Ablations

We use the hyperparameters described in Sec. 4.1.2 as the default setting for all experiments in the ablation unless specified otherwise. For AS on COIN we randomly sample 128 consecutive frames (instead of 512) for efficiency during training for the ablations. For the ablations we report ANet TAL with mAP@0.5IoU, ANet MR with Recall@1 under IoU=0.5 and COIN with mAP.

Architectural design choices. In Table 1, we ablate three design choices: the loss function, feature pyramid design, and the number of convolution layers in the localization heads. All losses perform similarly with L1 being slightly better than other three. ViTDet-style feature pyramid outperforms a standard FPN [35] as it introduces less additional layers to the pretrained model. Removing the feature pyramid completely significantly degrades the performance, with a 7.4% drop. Performance increases as we increase the number of convolution layers but saturates at 3. The best setup derived here is used by following experiments.

Variations on the text encoder and tokens. In Table 2, we freeze the CLIP image encoder, pair it with different text encoders, and finetune them. Using "unpaired" image-text encoders indeed diminishes the performance on all three tasks, especially for TAL and MR. For closed-vocabulary tasks, such as TAL, a text encoder is not strictly required. We hence compare our model to a version without the text encoder, and try to make minimum changes to ensure a fair comparison. Without the text tokens the video-text fusion module becomes a temporal encoder (*i.e.* a transformer which operates on frame-level features, aggregating temporal information across them). To enable this ablation, we also modify the linear projections in Eqs. 1 and 2 as follows:

$$\hat{\mathbf{Y}} = \mathbf{Z}\mathbf{W}_{cls} + \mathbf{b}_{cls}$$
$$\Delta\hat{\mathbf{T}} = \text{relu}(\mathbf{Z}\mathbf{W}_{reg} + \mathbf{b}_{reg})$$

where $\mathbf{Z} \in \mathbb{R}^{N\times K}$ are the activiations after convolution layers, $\hat{\mathbf{Y}} \in \mathbb{R}^{N\times C}$ and $\Delta\hat{\mathbf{T}} \in \mathbb{R}^{N\times 2C}$ are the predicted class logits and start/end time displacements.

After removing the text encoder, the performance on ANet TAL drops from 54.7 mAP@0.5IoU to 46.5 (a relative decrease of **15%**). In a second study, we also compare the performance of using a single text `[CLS]` token versus using all the text tokens from the text encoder on different tasks shown in Table 3. For close-vocabulary tasks, such as TAL and AS, *all* refers to 16 tokens to represent the class labels and for MR we increase the sequence length to *32*, i.e., captions contain more words than class labels. We demonstrate that using all tokens gives better performance on all tasks and such improvement is larger for tasks involved more complex language queries, such as MR.

Text Encoder	MParams	ANet TAL	ANet MR	COIN
T5-S	147.1	46.7	39.7	16.1
T5-B	221.5	46.6	39.9	15.9
CLIP-B	174.9	**53.3**	**44.2**	**16.4**

Table 2. **Effect of different text encoders**. We use the same frozen CLIP-B image encoder, with both T5 and CLIP-B text encoders and show results across all tasks. Paired image/text encoders significantly outperform unpaired encoders for localization tasks. Note that for COIN, results are reported using mAP.

# tokens	ANet TAL	ANet MR	COIN
All	**53.7**	**44.2**	**16.4**
One	53.3	42.6	15.7

Table 3. **Effect of number of text tokens.** We show that using all text tokens (16 tokens for both TAL and AS and 32 tokens for MR) performs better than using a single token in video-text fusion on different tasks. Note that the image encoder is frozen.

Effect of video-text fusion module. We also compare our model with a late-fusion variant where the frame relevancy scores are computed as the dot product between the normalized $\mathbf{Z}$ and the class label text embeddings. This variant improves over the no-text variant to 49.8 on ANet TAL but still worse than our proposed mid-fusion model. We find that video-text fusion is essential for achieving good performance on TAL.

Finetuning strategies. Table 4 compares four different strategies for finetuning a Kinetics-pretrained model on downstream tasks by either freezing or finetuning each of the two encoders. In this study, we always finetune the video-text fusion layers and heads. We observe that it is more beneficial to finetune the image encoder for close-vocabulary tasks, such as TAL and AS. However, for task involving more complex queries, such as MR, finetuning the image encoder actually degrades the performance. A similar phenomenon is also observed by [77], and may be due to overfitting.

4.3. Comparison with the state-of-the-art

In this section we compare to the state-of-the-art for all three tasks individually. Qualitative examples for each task are provided in Fig. 3.

Image/Text encoders	ANet TAL	ANet MR	COIN
frozen/frozen	53.2	43.4	16.1
frozen/finetuned	53.3	**44.2**	16.4
finetuned/frozen	**54.7**	39.7	16.6
finetuned/finetuned	54.3	41.2	**16.9**

Table 4. **Effect of freezing or finetuning image/text encoder on different tasks.** The video-text fusion module and heads are always finetuned. For closed-vocabulary tasks, such as TAL and AS, finetune the image encoder is better (bottom two rows), however, for tasks involving more complex queries such as MR, finetuning the image encoder degrades performance (top two rows).

	Method	Vision Enc.	R@1		R@5	
			IoU=0.5	IoU=0.7	IoU=0.5	IoU=0.7
Charades-STA	CTRL [16]	C3D	23.6	8.9	58.9	29.5
	2D TAN [81]	VGG	39.7	23.3	80.3	51.3
	VSLNet [80]	I3D	47.3	30.2	-	-
	UMT [40]	VGG	49.4	26.2	**89.4**	55.0
	IVG-DCL [50]	C3D	50.2	32.9	-	-
	M-DETR [30]	CLIP	55.7	34.2	-	-
	LGI [48]	I3D	59.5	35.5	-	-
	UnLoc-B	CLIP	58.1	35.4	87.4	**59.1**
	UnLoc-L	CLIP	**60.8**	**38.4**	88.2	**61.1**
ANet MR	LGI [48]	C3D	41.5	23.1	-	-
	VSLNet [80]	I3D	43.2	26.2	-	-
	2D TAN [81]	C3D	44.5	26.5	77.1	62.0
	DRN [76]	C3D	45.5	24.4	78.0	50.3
	VLG [60]	C3D	46.3	29.8	77.2	**63.3**
	UnLoc-B	CLIP	**48.0**	29.7	**81.5**	61.4
	UnLoc-L	CLIP	**48.3**	**30.2**	**79.2**	61.3
QVHighlights	M-DETR [30]	SF+CLIP	53.9	34.8	-	-
	UMT [40]	SF+CLIP	60.3	44.3	-	-
	QD-DETR [47]	SF+CLIP	62.4	45.0	-	-
	UnLoc-B	CLIP	**64.5**	**48.8**	-	-
	UnLoc-L	CLIP	**66.1**	**46.7**	-	-

Table 5. **Comparison with the state-of-the-art for Moment Retrieval.** We show results on Charades-STA (test split), ANet MR (val_2 split), and QVHighlights (val split) datasets.

Moment retrieval For MR models we freeze the image encoder and finetune the rest of the network following the best strategy derived in Table 4. On ANet MR, our UnLoc-L model achieves a new state-of-the-art improving the previous best by 2.0% and 0.4% in recall@1 under IoU=0.5 and 0.7, respectively (Table 5). On Charades-STA, our UnLoc-L model improves upon the previous best [48] by 1.3% and 2.9% on the same two metrics. On ANet MR, UnLoc-L outperforms [48] by a larger margin, 6.8% and 7.1%. On QVHighlights, UnLoc- L improves upon the previous best [47] by 3.7% and 1.7%. Most previous work is built upon pre-extracted convolutional features, such as I3D [7], P3D [54], C3D [64], R(2+1)D [65], VGG [59], SlowFast [14], etc, and our work is most comparable to [30], which also employs CLIP features (in addition to SlowFast [14] features). Our UnLoc-L model scores 5.1% and 4.4% higher than [30] on Charades-STA in recall@1 under IoU=0.5 and 0.7. To the best of our knowledge, we are the first work employing pure transformer features that achieves state-of-the-art results on moment retrieval, which has largely been dominated by CNN-based features.

Setting	Method	Vision Encoder	mAP@0.5IoU
Finetuned	A2Net [74]	I3D	43.6
	TSP [1]	R(2+1)D	51.3
	GTAN [41]	P3D	52.6
	VSGN [82]	I3D	53.3
	TadTR [39]	R(2+1)D	53.6
	PBRNet [38]	I3D	54.0
	TCANet [53]	SlowFast	54.3
	ActionFormer [79]	R(2+1)D	54.7
	ContextLoc [87]	I3D	56.0
	EffPrompt [28]	CLIP	44.0
	STALE [49]	CLIP	54.3
	STALE [49]	I3D	56.5
	UnLoc-B (1st prompt)	CLIP	54.6
	UnLoc-L (1st prompt)	CLIP	**58.8**
	UnLoc-L (prompt ensembling)	CLIP	**59.3**
Zero-shot 50% Seen 50% Unseen	EffPrompt [28]	CLIP	32.0
	STALE [49]	CLIP	32.1
	UnLoc-B (1st prompt)	CLIP	**36.9**
	UnLoc-L (1st prompt)	CLIP	**43.2**
	UnLoc-L (prompt ensembling)	CLIP	**43.7**
Zero-shot 75% Seen 25% Unseen	EffPrompt [28]	CLIP	37.6
	STALE [49]	CLIP	38.2
	UnLoc-B (1st prompt)	CLIP	**40.2**
	UnLoc-L (1st prompt)	CLIP	**47.4**
	UnLoc-L (prompt ensembling)	CLIP	**48.8**

Table 6. **Comparison with the state-of-the-art on ANet TAL.** We show results for finetuning, and both the zero-shot (open-set) protocols introduced by [28]. Our method outperforms all previous work across all settings, achieving strong gains particularly in the zero-shot settings.

Temporal localization Table 6 shows results on ANet TAL under two settings (finetuned and zero-shot). In the finetuned setting, we freeze the text encoder and finetune the rest of the network following the best strategy derived from Table 4. For UnLoc-L we increase the sampled frames to 160 and use a 5-L Feature Pyramid. As shown in Table 6, most high-performance methods are built on top of 3D convolutional features. There are two previous attempts to replace the CNN vision encoder by a Transformer encoder. EffPrompt [28], built on top of frozen CLIP features, scored significantly lower than recent CNN-based models and STALE [49], which is also built upon CLIP features, achieved competitive results with the best CNN methods but is 2.2 worse than the same model trained on two-stream I3D features. To the best of our knowledge, we are the first work that achieved state-of-the-art results using only Transformer features. Our UnLoc-L model improved previous best results in terms of mAP@0.5IoU by 2.3 and with prompt ensembling this margin is increased to 2.8.

For both splits in the zero-shot (open-set) protocols proposed by [28, 49], UnLoc-B and L outperform previous best by a significant margin. Specifically, UnLoc-L advances

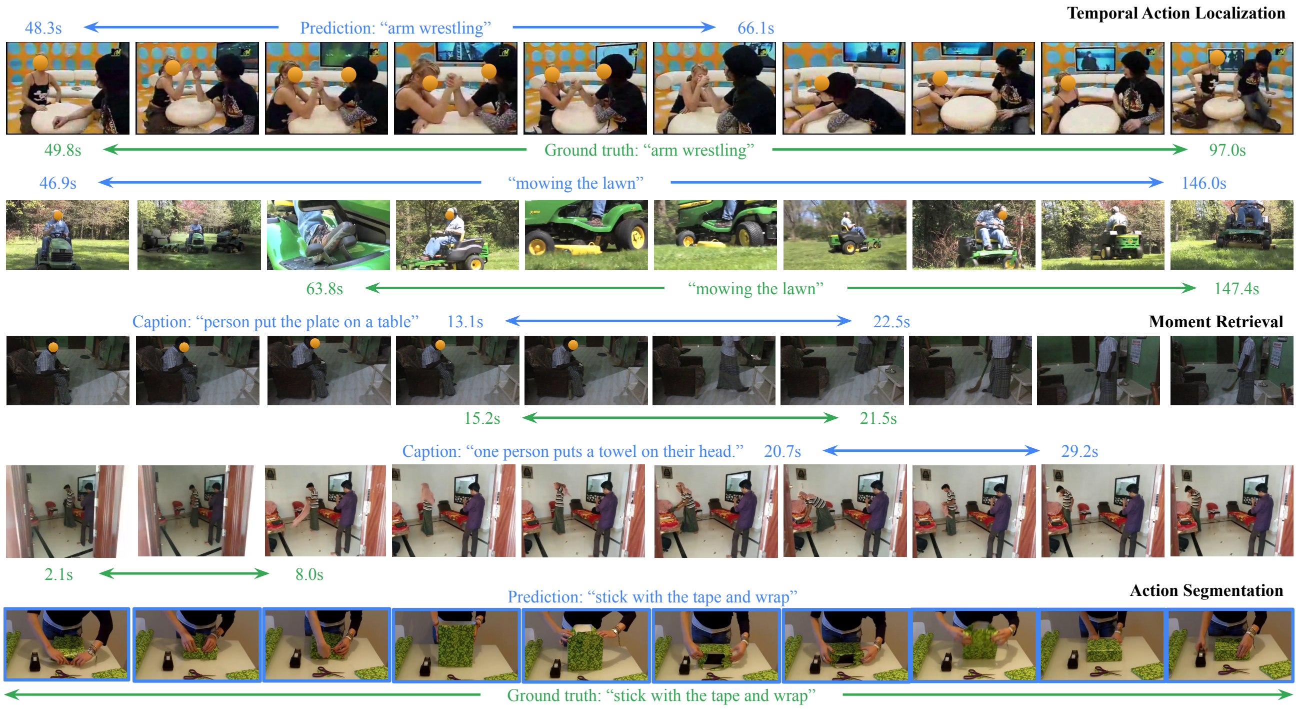
Temporal Action Localization
48.3s
Prediction: "arm wrestling"
66.1s
49.8s
Ground truth: "arm wrestling"
97.0s
46.9s
"mowing the lawn"
146.0s
63.8s
"mowing the lawn"
147.4s
Caption: "person put the plate on a table"
13.1s
22.5s
Moment Retrieval
15.2s
21.5s
Caption: "one person puts a towel on their head."
20.7s
29.2s
2.1s
8.0s
Prediction: "stick with the tape and wrap"
Action Segmentation
Ground truth: "stick with the tape and wrap"

Figure 3. **Qualitative Results** We show results on ActivityNet, Charades and COIN, for Temporal Action Localization, Moment Retrieval and Action Segmentation respectively. Predictions are shown in blue, while the ground truth is in green (best viewed in colour). For action segmentation, the ground truth covers the entire clip. Note how our model is able to predict accurate boundaries, in some cases better refined than the ground truth (top row, the arm wrestling action has stopped, however the ground truth boundary extends for a while after). For the second example for Moment Retrieval (4th row from top), we show a failure case, where our model detects the moment where the towel is 'put down', and not 'on their head' as perhaps the latter is a rarer occurrence in the training data.

Method	Frame accuracy	mAP
Baseline: predict all background	58.9	0.0
ActBERT [86]	57.0	-
MIL-NCE [44]	61.0	-
TACo [73]	68.4	-
VLM [69]	68.4	-
VideoCLIP [70]	68.7	-
UniVL [42]	70.0	-
UnLoc-B (1st prompt)	68.0	36.2
UnLoc-L (1st prompt)	**72.6**	47.0
UnLoc-L (prompt ensembling)	**72.8**	47.7

Table 7. **Comparison with the state-of-the-art on COIN for Action Segmentation.** We report results using both frame accuracy (as is standard practice) and mAP, which we believe is a better metric given that a large proportion (58.9%) of the dataset is labelled as a single class (background).

previous state-of-the-art by 11.6, a relative 36.1% improvement on the 50/50 split and by 10.6, a relative 27.7% on the 75/25 split.

Action segmentation Table 7 compares our model with previous work and UnLoc-L outperform previous state-of-the-art by 2.8% in frame accuracy. Besides architectural differences, we note that previous works [44, 70, 42] pretrain their models on HowTo100M [45], which consists of around 100M aligned ASR and video clip pairs, and is also in a similar domain to COIN (instructional web videos). Our models on the other hand, are initialized from CLIP checkpoints, which are trained on cleaner web image-text pairs from multiple domains and finetuned on Kinetics, 10s clips of human activity videos.

5. Conclusion and Future Work

We propose a new model for video localization tasks, called UnLoc. UnLoc consists of a two-tower CLIP model, the output features of which are fed into a video-text fusion module and feature pyramid. Unlike previous works, we achieve state-of-the-art results on 3 different benchmarks (moment retrieval, temporal action localization and action segmentation) with a single approach, without the need for action proposals or pretrained video features.

Future work will investigate cotraining on the three localization tasks, pretraining on large, weakly labelled datasets, exploring highlight detection as an additional downstream task, and adapting our model to other modalities such as audio for sound localization [25].

References

[1] Humam Alwassel, Silvio Giancola, and Bernard Ghanem. Tsp: Temporally-sensitive pretraining of video encoders for localization tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3173–3183, 2021. 7

[2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. ViViT: A Video Vision Transformer. In *ICCV*, 2021. 5

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. In *arXiv preprint arXiv:1607.06450*, 2016. 4

[4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. A clip-hitchhiker's guide to long video retrieval. *arXiv preprint arXiv:2205.08508*, 2022. 1, 2

[5] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms–improving object detection with one line of code. In *ICCV*, pages 5561–5569, 2017. 4

[6] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019. 5

[7] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 2, 7

[8] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A. Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster r-cnn architecture for temporal action localization. In *CVPR*, 2018. 2

[9] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. Temporally grounding natural sentence in video. In *EMNLP*, 2018. 2

[10] Long Chen, Chujie Lu, Siliang Tang, Jun Xiao, Dong Zhang, Chilie Tan, and Xiaolin Li. Rethinking the bottom-up framework for query-based video localization. In *AAAI*, 2020. 2

[11] Ekin D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. Randaugment: Practical automated data augmentation with a reduced search space. In *NeurIPS*, 2020. 5

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 3

[13] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[14] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 2, 7

[15] Kunihiko Fukushima. Cognitron: A self-organizing multilayered neural network. *Biological cybernetics*, 20(3-4):121–136, 1975. 4

[16] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, pages 5267–5275, 2017. 1, 2, 5, 7

[17] Junyu Gao and Changsheng Xu. Fast video moment retrieval. In *ICCV*, 2021. 2

[18] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, 2022. 1

[19] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, 2022. 2

[20] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *CVPR*, 2019. 2

[21] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, pages 961–970. IEEE, 2015. 1, 2, 5

[22] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017. 2

[23] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with temporal language. In *EMNLP*, 2018. 2

[24] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Weinberger. Deep networks with stochastic depth. In *ECCV*, 2016. 5

[25] Jaesung Huh, Jacob Chalk, Evangelos Kazakos, Dima Damen, and Andrew Zisserman. Epic-sounds: A large-scale dataset of actions that sound. *arXiv preprint arXiv:2302.00646*, 2023. 8

[26] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017. 1, 2

[27] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 1

[28] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. In *ECCV*, pages 105–124. Springer, 2022. 1, 2, 5, 7

[29] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, pages 706–715, 2017. 1, 2, 4

[30] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. In *NeurIPS*, 2021. 2, 5, 7

[31] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *ECCV*, pages 280–296. Springer, 2022. 2, 4

[32] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *ICCV*, 2019. 2, 5

[33] Tianwei Lin, Xu Zhao, and Zheng Shou. Single shot temporal action detection. *Proceedings of the 25th ACM international conference on Multimedia*, Oct 2017. 2

[34] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018. 2, 5

[35] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 2117–2125, 2017. 4, 6

[36] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 4

[37] Ziyi Lin, Shijie Geng, Renrui Zhang, Peng Gao, Gerard de Melo, Xiaogang Wang, Jifeng Dai, Yu Qiao, and Hongsheng Li. Frozen clip models are efficient video learners. In *ECCV*, 2022. 1, 2

[38] Qinying Liu and Zilei Wang. Progressive boundary refinement network for temporal action detection. In *AAAI*, volume 34, pages 11612–11619, 2020. 7

[39] Xiaolong Liu, Qimeng Wang, Yao Hu, Xu Tang, Shiwei Zhang, Song Bai, and Xiang Bai. End-to-end temporal action detection with transformer. *IEEE Transactions on Image Processing*, 31:5427–5441, 2022. 7

[40] Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection. In *CVPR*, pages 3042–3051, 2022. 2, 5, 7

[41] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Gaussian temporal awareness networks for action localization. In *CVPR*, pages 344–353, 2019. 7

[42] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020. 2, 8

[43] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. CLIP4Clip: An empirical study of clip for end to end video clip retrieval. In *arXiv:2104.08860*, 2021. 2

[44] 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. 2, 8

[45] 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, 8

[46] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple Open-Vocabulary Object Detection with Vision Transformers. In *ECCV*, 2022. 1, 2

[47] WonJun Moon, Sangeek Hyun, SangUk Park, Dongchan Park, and Jae-Pil Heo. Query-dependent video representation for moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23023–23033, 2023. 7

[48] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-global video-text interactions for temporal grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10810–10819, 2020. 1, 2, 7

[49] Sauradip Nag, Xiatian Zhu, Yi-Zhe Song, and Tao Xiang. Zero-shot temporal action detection via vision-language prompting. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 681–697. Springer, 2022. 1, 2, 5, 7

[50] Guoshun Nan, Rui Qiao, Yao Xiao, Jun Liu, Sicong Leng, Hao Zhang, and Wei Lu. Interventional video grounding with dual contrastive learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2765–2775, 2021. 2, 7

[51] Megha Nawhal and Greg Mori. Activity graph transformer for temporal action localization. In *arXiv 2101.08540*, 2021. 2

[52] Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. Expanding language-image pretrained models for general video recognition. In *ECCV*, 2022. 1, 2

[53] Zhiwu Qing, Haisheng Su, Weihao Gan, Dongliang Wang, Wei Wu, Xiang Wang, Yu Qiao, Junjie Yan, Changxin Gao, and Nong Sang. Temporal context aggregation network for temporal action proposal refinement. In *CVPR*, pages 485–494, 2021. 7

[54] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*, pages 5533–5541, 2017. 7

[55] 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. In *ICML*, 2021. 1, 3, 5

[56] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? In *NeurIPS*, 2021. 4

[57] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*, 2016. 2

[58] Bernard Ghanem Shyamal Buch, Victor Escorcia and Juan Carlos Niebles. End-to-end, single-stream temporal action detection in untrimmed videos. In *BMVC*, pages 93.1–93.12, 2017. 2

[59] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 2, 7

[60] Mattia Soldan, Mengmeng Xu, Sisi Qu, Jesper Tegner, and Bernard Ghanem. Vlg-net: Video-language graph matching network for video grounding. In *ICCV*, pages 3224–3234, 2021. 1, 2, 4, 7

[61] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer. In *arXiv 1906.05743*, 2019. 2

[62] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *CVPR*, pages 1207–1216, 2019. 1, 2, 5

[63] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training

data-efficient image transformers & distillation through attention. In *arXiv preprint arXiv:2012.12877*, 2020. 5

[64] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015. 2, 7

[65] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018. 2, 7

[66] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3

[67] Jingwen Wang, Lin Ma, and Wenhao Jiang. Temporally grounding language queries in videos by contextual boundary-aware prediction. In *AAAI*, 2019. 2

[68] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. In *arXiv:2109.08472*, 2021. 2

[69] Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. VLM: Task-agnostic video-language model pre-training for video understanding. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4227–4239, Online, Aug. 2021. Association for Computational Linguistics. 8

[70] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *EMNLP*, 2021. 2, 8

[71] Mengmeng Xu, Chen Zhao, David S. Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *CVPR*, 2020. 5

[72] Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. *arXiv preprint arXiv:2302.14115*, 2023. 2

[73] Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. Taco: Token-aware cascade contrastive learning for video-text alignment. In *ICCV*, pages 11562–11572, 2021. 8

[74] Le Yang, Houwen Peng, Dingwen Zhang, Jianlong Fu, and Junwei Han. Revisiting anchor mechanisms for temporal action localization. *IEEE Transactions on Image Processing*, 29:8535–8548, 2020. 7

[75] Yitian Yuan, Lin Ma, Jingwen Wang, Wei Liu, and Wenwu Zhu. Semantic conditioned dynamic modulation for temporal sentence grounding in videos. In *Advances in Neural Information Processing Systems*, pages 534–544, 2019. 2

[76] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *CVPR*, pages 10287–10296, 2020. 2, 4, 7

[77] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18123–18133, 2022. 6

[78] Bowen Zhang, Hexiang Hu, Joonseok Lee, Ming Zhao, Sheide Chammas, Vihan Jain, Eugene Ie, and Fei Sha. A hierarchical multi-modal encoder for moment localization in video corpus. In *arXiv 2011.09046*, 2020. 2

[79] Chenlin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In *ECCV*, 2022. 2, 4, 5, 7

[80] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. Span-based localizing network for natural language video localization. In *ACL*, 2020. 2, 7

[81] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2d temporal adjacent networks for moment localization with natural language. In *AAAI*, 2020. 2, 7

[82] Chen Zhao, Ali K Thabet, and Bernard Ghanem. Video self-stitching graph network for temporal action localization. In *ICCV*, pages 13658–13667, 2021. 7

[83] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *ICCV*, 2017. 2

[84] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. Distance-iou loss: Faster and better learning for bounding box regression. In *AAAI*, 2020. 4

[85] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16793–16803, 2022. 2

[86] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, pages 8746–8755, 2020. 8

[87] Zixin Zhu, Wei Tang, Le Wang, Nanning Zheng, and Gang Hua. Enriching local and global contexts for temporal action localization. In *ICCV*, pages 13516–13525, 2021. 2, 7