# Test-Time Zero-Shot Temporal Action Localization

Benedetta Liberatori[1]    Alessandro Conti[1]    Paolo Rota[1]    Yiming Wang[2]    Elisa Ricci[1,2]

University of Trento[1]    Fondazione Bruno Kessler[2]

https://github.com/benedettaliberatori/T3AL

## Abstract

*Zero-Shot Temporal Action Localization (ZS-TAL) seeks to identify and locate actions in untrimmed videos unseen during training. Existing ZS-TAL methods involve fine-tuning a model on a large amount of annotated training data. While effective, training-based ZS-TAL approaches assume the availability of labeled data for supervised learning, which can be impractical in some applications. Furthermore, the training process naturally induces a domain bias into the learned model, which may adversely affect the model's generalization ability to arbitrary videos. These considerations prompt us to approach the ZS-TAL problem from a radically novel perspective, relaxing the requirement for training data. To this aim, we introduce a novel method that performs Test-Time adaptation for Temporal Action Localization (T3AL). In a nutshell, T3AL adapts a pre-trained Vision and Language Model (VLM). T3AL operates in three steps. First, a video-level pseudo-label of the action category is computed by aggregating information from the entire video. Then, action localization is performed adopting a novel procedure inspired by self-supervised learning. Finally, frame-level textual descriptions extracted with a state-of-the-art captioning model are employed for refining the action region proposals. We validate the effectiveness of T3AL by conducting experiments on the THUMOS14 and the ActivityNet-v1.3 datasets. Our results demonstrate that T3AL significantly outperforms zero-shot baselines based on state-of-the-art VLMs, confirming the benefit of a test-time adaptation approach.*

## 1. Introduction

Zero-shot Temporal Action Localization (ZS-TAL) aims to locate and recognize actions in any video sequence, enabling the recognition of classes unseen during training. Large-scale Vision and Language models (VLMs) [1, 12, 22, 28, 31] are renowned for the exceptional generalization capability derived from their extensive pre-training on web-scale image-text datasets, outperforming traditional image classification methods [4, 13, 32]. When applied to
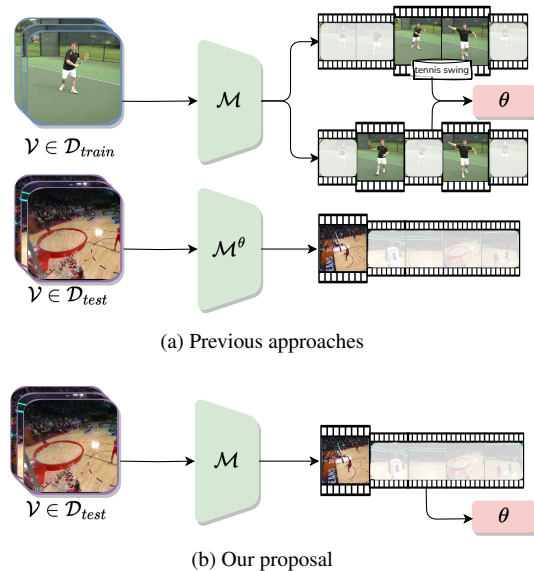


(a) Previous approaches



(b) Our proposal

Figure 1. **Task setup.** Previous approaches tackling ZS-TAL (a) train the model on labelled data and test it in-domain. Due to lack of out-of-distribution generalization, we propose to update the parameters at test-time on a stream of unlabelled videos without prior supervised training (b).

the video domain, however, VLMs typically require fine-tuning to account for the image-video structural domain shift [11, 26, 28].

Recent methods exploiting VLMs for ZS-TAL also abide by this limitation, requiring training data to learn the video domain and localize unseen actions at test time [9, 20, 30] (see Fig. 1(a)).

While model fine-tuning has the clear objective of learning video representations, which allows to effectively localize actions in the untrimmed videos, it also assumes the availability of a large annotated data collection. In certain applications, however, such datasets may be unavailable. Furthermore, fine-tuning inherently carries the downside of producing models with decreased out-of-domain generalization capabilities [35]. This latter issue is especially severe for ZS-TAL. A preliminary investigation (see Sec. 3) demon-

strates that state-of-the-art ZS-TAL evaluated in a cross-domain setting suffer from a dramatic drop in performance. This clearly raises concerns regarding the adaptability and robustness of existing ZS-TAL approaches, expecially in real-world scenarios where training data is inaccessible due to privacy concerns or when a major data distribution shift occurs over time.

Motivated by these observations, in this work we propose to investigate the problem of ZS-TAL under a novel perspective, featuring the relevant scenario where training data is inaccessible. Even with the aid of powerful VLMs, locating and recognizing actions without training is undoubtedly challenging, since videos carry additional complexities with respect to images, induced by the scene clutter and the difficulty of modelling the temporal dynamics [19]. Nonetheless, we argue that, even in the absence of training data, videos made available at inference time can be exploited as a rich source of information for temporal action localization.

Inspired by recent works on Test-Time Adaptation (TTA) [24, 25], we propose $\mathbf{T}^3\mathbf{AL}$, standing for **T**est **T**ime adaptation for **T**emporal **A**ction **L**ocalization. $T3AL$ utilizes a pre-trained VLM not fine-tuned on training data, as opposed to previous works [9, 20, 30]. Our solution instead, undergoes direct adaptation when a video sequence is made available during inference (Fig. 1(b)). $T3AL$ unfolds in three key steps. First, we compute video-level pseudo-labels corresponding to action categories by aggregating semantic information extracted from each frame by the VLM image encoder. Next, a first solution to temporal action localization is produced, based on the results from video pseudo-labels by employing a novel procedure inspired by self-supervised learning [5]. The computed video action proposals are then refined with frame-level textual descriptions extracted from a state-of-the-art captioning model. We evaluate $T3AL$ on two publicly available benchmarks, *i.e.* THUMOS14 [7] and ActivityNet-v1.3 [6], achieving a relative improvement of $+6.3\%$ and $+13.5\%$ when compared to a naive application of VLMs for TAL without TTA. Through several oracle experiments, we also demonstrate the existence of a potential space for future improvements. This suggests that test-time ZS-TAL is a viable approach for locating and recognising actions in arbitrary video sequences. We hope that these findings will inspire future research in this area.

Our contributions can be summarized as follows:

- We address ZS-TAL in a new practical scenario where training data is unavailable. We demonstrate that this is a challenging problem as state-of-the-art methods for the task poorly generalize without training.
- We propose $T3AL$, the first method that tackles ZS-TAL without training data by leveraging a pre-trained VLM. $T3AL$ benefits from an effective TTA strategy and from external knowledge derived from generated captions.
- We empirically demonstrate that adapting on an unlabeled

stream of data is a viable solution to the out-of-distribution issue of current training-based approaches for ZS-TAL.

## 2. Related work

Our work is closely related to existing literature in Zero-Shot Temporal Action Localization and Test-Time Adaptation (TTA), which we briefly review in the current section.

**Zero-Shot Temporal Action Localization.** Temporal Action Localization (TAL) methods jointly perform action localization and recognition. Existing works either tackle the problems sequentially, *i.e.*, two-stage [3, 9, 27, 34], or concurrently, *i.e.*, one-stage [2, 15, 20, 30]. Two-stage methods first identify class-agnostic region proposals and then classify each region. One-stage methods perform action localization and classification simultaneously. Traditionally, both one-stage and two-stage approaches work in closed-set scenarios, where train and test data share the same action categories. Recently, EffPrompt [9] introduced the novel ZS-TAL setup, removing the above premise and isolating action categories between training and testing. To address the novel setup, EffPrompt employs a two-stage architecture to generate action proposals with an off-the-shelf detector [14] and then classifies action proposals with CLIP [22]. Differently, STALE [20] proposes to train a CLIP-based proposal-free model using two concurrent streams for localization and classification. The localization branch focuses on learning a class-agnostic representation masking, while the classification stream aligns the masked features with the text embeddings of the respective class, contributing to the final classifier output. More recently, UnLoc [30] extracts joint features for video-text pairs with CLIP and feeds them into a dedicated fusion module. A feature pyramid architecture then takes these refined outputs and establishes hierarchical connections, predicting per-frame relevance scores and start/end time displacements.

Although effective, existing ZS-TAL methods require learning a model on a training set, leading to several inherent limitations. As discussed above, these limitations encompass challenges in generalization to out-of-domain data, high computational requirements and reliance on availability of annotated data. We tackle a more practical yet challenging setup for ZS-TAL where the training set is inaccessible. Our method falls into the one-stage category, addressing the challenges of ZS-TAL in an integrated manner.

**Test-Time Adaptation.** In TTA, a model pre-trained on a training dataset must adapt to an unknown test distribution expressed as an unlabelled data stream [24, 25]. Several works propose TTA methods for image classification. For instance, TENT [25] adapts a pre-trained network at test-time by minimizing the entropy of the batch-wise prediction probability distributions. Similarly, MEMO [33] reduces the entropy of the marginal distribution across augmentations, overcoming the necessity for multiple samples. Recent works

explore these concepts for large-scale VLMs [17, 18, 23]. TPT [18] adapts CLIP by learning textual context vectors via entropy minimization. SwapPrompt [17] employs a swapping mechanism between the online prompt and its historical moving average to improve and stabilize adaptation. PromptAlign [23] fine-tunes multi-modal prompts at test-time by aligning the distribution statistics obtained from multiple augmented views of a single test image with the training data distribution statistics.

While all these methods tackle TTA in the image domain, videos still remains largely unexplored. A notable exception is ViTTA [16], which performs test-time adaptation for video action recognition, handling the distribution shifts and aligning test and pre-computed train statistics online. Another contribution in this area is RNA$^{++}$ [21] which proposes a TTA approach to address the problem of domain shift in egocentric action recognition. This is particularly relevant when unsupervised domain adaptation approaches, while effective, are impractical due to the unavailability of data from the target distribution. Our research differs significantly from [16, 21], as we adapt models not specifically designed to resolve the TAL task. This necessitates the model to deduce a previously unencountered task.

## 3. Cross-dataset generalization analysis

We propose a preliminary experiment to motivate further the novel research direction proposed in this work. The goal is to test the generalization capability of state-of-the-art methods for ZS-TAL, testing their action localization performance in a cross-dataset setting.

Specifically, we consider two state-of-the-art methods, EffPrompt [9] and STALE [20][1]. For EffPrompt [9], we use their off-the-shelf detector [14] and their action recognition model trained on HMDB51 [10]. For STALE [20], we use their ZS-TAL model trained on ActivityNet-v1.3. We conduct both experiments using THUMOS14 as a target dataset. Note that our *cross-domain* protocol evaluates these models, which are trained on more challenging and diverse datasets (*i.e.*, ActivityNet-v1.3, 200 classes; HMDB51, 51 classes), on a simpler data collection (*i.e.*, THUMOS14, 20 classes). For reference, we also report results obtained in an *in-domain* setting, *i.e.* where both methods are trained and tested on THUMOS14.

The results of our preliminary investigation show that EffPrompt and STALE do not generalize on out-of-distribution samples, despite i) being trained to improve their zero-shot capabilities (i.e., their ability to recognize unseen classes) and ii) the usage of prior knowledge encoded in VLMs. Fig. 2 shows the comparison between the performance of the two methods in a *in-domain* and a *cross-domain* setting. The plot shows a drastic drop (*i.e.*, higher than 15%

---

[1]No public code implementation was available for UnLoc [30] at the time of the submission.
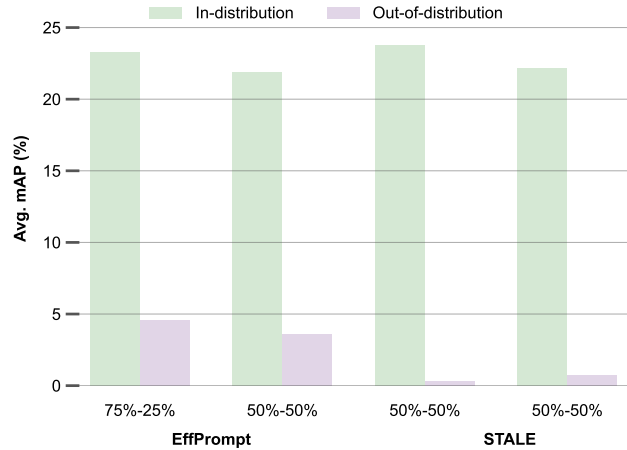


Figure 2. **Cross-dataset generalization.** We show the average mAP, computed at IoU thresholds of [0.3:0.1:0.7], for EffPrompt and STALE trained and tested on THUMOS14 , and trained on a different dataset and tested on THUMOS14. We report results for the 75:25 (75% seen classes) and 50:50 (50% seen classes) evaluation settings.

mAP in both settings) in the performance of both methods when the test dataset differs with respect to the one used for model fine-tuning. We associate this behavior to the fact that a perturbation of the weights of the VLM boosts *in-domain* prediction ability, but hinders *out-domain* generalization. Motivated by these experimental findings, we devise a method to achieve robust performance on different datasets without the need for annotated training data.

## 4. T$^3$AL: Test Time Adaptation for Temporal Action Localization

In this section we first define the problem, then we detail the main steps of $T3AL$: video-level pseudo-labelling, self-supervised prediction refinement and text-guided region suppression.

### 4.1. Problem definition

A ZS-TAL algorithm aims to identify and classify actions in untrimmed videos. For each detected temporal region, the model predicts the class and indicate when it starts and ends. While the set of classes is given, they differ from the categories seen by the model at training. Existing literature addressing ZS-TAL [9, 20, 30] always involves a labelled training set $\mathcal{D}_{train}$ and a test set $\mathcal{D}_{test}$, with two disjoint sets of action classes. Yet, as shown in Sec. 3, state-of-the-art methods greatly rely on $\mathcal{D}_{train}$, resulting in poor generalization if the two datasets are drawn from different distributions. In this paper, we advocate for the need to investigate a different scenario for ZS-TAL—relevant for practical applications—where *in-domain* training data is unavailable.
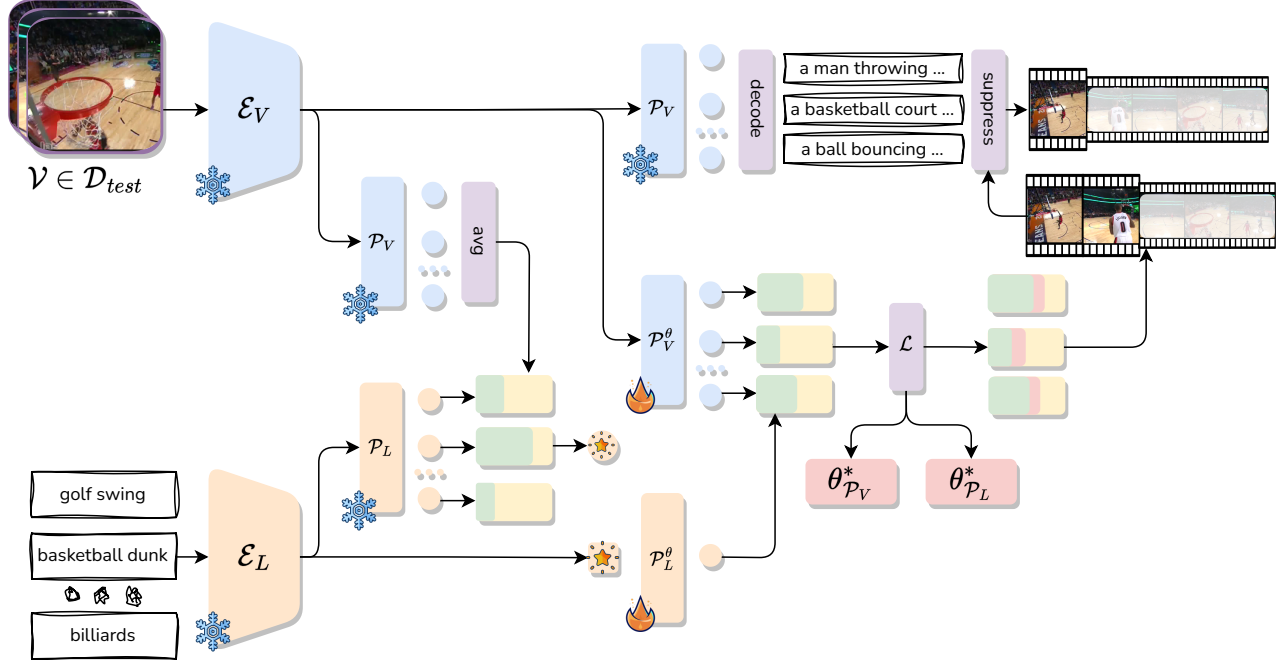
Figure 3. **Overview of the proposed method.** $T3AL$ addresses the task of ZS-TAL by only learning at test-time on unlabelled data. We first compare the average visual frames with the textual class names to identify the video pseudo-label ✦. We then refine the visual frames - video pseudo-label scores with self-supervision. Last, we exploit the decoder of a captioning model (*i.e.*, CoCa [31]) to generate captions and perform text-guided region suppression . We only 🔥 fine-tune the vision and language projectors, while keeping the encoders ❄ frozen. Once the prediction is obtained, the optimized parameters $\theta^*_{\mathcal{P}_V}$ and $\theta^*_{\mathcal{P}_L}$ are re-initialized to the ones of the pre-trained model.

Given a video $\mathcal{V}$, our goal is to localize regions and assign corresponding actions from a set of classes $\mathcal{C}$, without having access to $\mathcal{D}_{train}$. The $M$ predicted action proposals are defined as $\{(y_i, t_i)\}_{i=1}^{M}$, where $y_i \in \mathcal{C}$ is the class and $t_i \in \mathbb{R}^2$ is the start/end time displacement of each action.

$T3AL$ is built on top of a pre-trained VLM model $\mathcal{M}$, consisting of a vision encoder $\mathcal{E}_V$ and a language encoder $\mathcal{E}_L$, as shown in Fig. 3. $T3AL$ directly addresses ZS-TAL at inference time only exploiting a single test video at a time. First, the frame-level representations extracted from the visual frames with $\mathcal{E}_V$ are averaged and compared with the class textual representations to identify the video pseudo-label. Then, the scores of the visual frames are computed and refined adapting $\mathcal{M}$ at test-time with self-supervision. Finally, we exploit a captioning model to generate captions and perform an additional suppression step. $T3AL$ is applied on a sample basis. Once the prediction for one sample is made, the optimized parameters of $\mathcal{M}$ are reverted to the original initialization.

### 4.2. Video-level pseudo-labelling

At test-time the only accessible information is the unlabelled sample $\mathcal{V} = \{x_i\}_{i=1}^{N}$ and the set of action categories $\mathcal{C}$. However, we can mitigate such lack of supervision by lever-

aging the knowledge already encoded in $\mathcal{M}$, as VLMs have demonstrated strong zero-shot capabilities in a wide range of classification tasks. First, we compute a compact representation from $\mathcal{V}$ as the average of its $N$ frames' latent representations extracted with the vision encoder:

$$\bar{\mathcal{V}} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{E}_V (x_i) \qquad (1)$$

In this compact representation the noise coming from non-informative frames present in the video is mitigated. Thus, we can exploit $\bar{\mathcal{V}}$ to compute a pseudo-label $y^{*2}$ for the whole video $\mathcal{V}$, selecting the label with the maximum cosine similarity:

$$y^* = \underset{y \in \mathcal{C}}{\mathrm{argmax}} \cos \left( \bar{\mathcal{V}}, \mathcal{E}_L (y) \right) \qquad (2)$$

where $\cos (\cdot, \cdot)$ indicates the cosine similarity. We propose to use $y^*$ to guide the localization process, providing a foundation for more temporally fine-grained predictions.

### 4.3. Self-supervised prediction refinement

The objective of the second step of our method is to refine this coarse-grained video prediction to effectively localize

---
[2]In Fig 3, the pseudo-label is encoded as a star.

regions in $\mathcal{V}$ where the action of class $y^*$ occurs. The video comprises frames that capture the pseudo-label $y^*$, alongside frames that do not exhibit any correspondence with it. The model $\mathcal{M}$ can easily classify these frames, but struggles on those lying in-between the two extremes as they involve visual cues semantically linked to $y^*$ while neglecting the actual execution of the action. Following this intuition, we propose to utilize those frames on which the model $\mathcal{M}$ is confident to filter out the noise in the prediction of the more ambiguous ones. Therefore, we compute the semantic closeness of every frame in the video to $y^*$, assigning the score:

$$s_i = \cos(\mathcal{E}_V(x_i), \mathcal{E}_L(y^*)). \qquad (3)$$

and for each frame in $\mathcal{V}$ we denote the corresponding feature representation as $z_i = \mathcal{E}_V(x_i)$.

Frames with higher scores are more likely to be associated with the foreground, $i.e.$, actions present in the video, while frames with lower scores are more likely to correspond to the background, representing non-action-related content. Building on this consideration, we aim to strategically leverage these frames with a self-supervised objective to refine the initial predictions. Specifically, we form a set of positive samples $\mathcal{Z}^+$ from features with higher scores, and a set of negative samples $\mathcal{Z}^-$ from features with lower scores:

$$\mathcal{Z}^+ = \{(z_i^+, s_i^+)\}_{i=1}^K, \quad \mathcal{Z}^- = \{(z_i^-, s_i^-)\}_{i=1}^K \quad (4)$$

For both sets, the $K$ features to be selected are distributed over the temporal dimension with a slight perturbation governed by a random noise $\epsilon$, avoiding concentration and ensuring diversity in the selection.

Our self-supervised objective can be formulated as:

$$\left(\theta_{\mathcal{P}_V}^*, \theta_{\mathcal{P}_L}^*, \tau^*\right) = \operatorname*{argmin}_{\theta_{\mathcal{P}}, \tau} \mathcal{L} \qquad (5)$$

where we only adapt the parameters of the two projections $\theta_{\mathcal{P}} = (\theta_{\mathcal{P}_V}, \theta_{\mathcal{P}_L})$, and the temperature parameter $\tau$.

The loss $\mathcal{L}$ can be further decomposed into two, with one taking as input the visual representations of the frames and the other taking as input the scores:

$$\mathcal{L} = \mathcal{L}_z + \mathcal{L}_s, \qquad (6)$$

where $\mathcal{L}_z$ is the Representation loss and $\mathcal{L}_s$ is the Separation loss. The Representation loss is exclusively applied to the positive set $\mathcal{Z}^+$ to bring positive frames closer in the embedding space. Differently, we refrain from applying the same objective to negative samples, as the background frames of the video carry much diverse information with various non-action-related visual cues. Without any guarantees of a shared semantics, we should not force the representations of these frames to be represented closer in space.

To address this, we adopt the BYOL [5] loss, commonly utilized in self-supervised learning, particularly in scenarios lacking negative examples.

This Representation loss enforces both visual and semantic closeness among potentially repeated instances of the same action within the video. Additionally, we assume that semantic knowledge is a continuous function, $i.e.$, the information in a frame at time $t$ is likely to be similar to that in adjacent frames, incorporating them into the set of positive candidates.

While the BYOL loss requires augmented views of a sample, we exploit the natural temporal dimension of videos to obtain multiple views for free. Following these observations, all aforementioned positive samples can be seen as augmented views and used to compute the loss as:

$$\mathcal{L}_z = 2 - 2 \cdot \frac{<z_i^+, z_j^+>}{\|z_i^+\|_2 \cdot \|z_j^+\|_2} \qquad (7)$$

where $z_i^+$ and $z_j^+$ are randomly sampled at each step.

The Separation loss is applied to both $\mathcal{Z}^+$ and $\mathcal{Z}^-$ and aims to push the scores of positive samples closer to 1 and the negative ones to 0, promoting their separation. It is again implemented as a BYOL loss (this component is ablated in Sec. 5.2). Specifically, we define the prediction vector as the concatenation of positive ad negative scores:

$$\mathbf{s} = \texttt{concat}\left(\{s_i^+\}_{i=1}^K, \{s_i^-\}_{i=1}^K\right) \in \mathbb{R}^{2K} \qquad (8)$$

and the binary target vector accordingly:

$$\mathbf{b} = \begin{bmatrix} 1_K \\ 0_K \end{bmatrix} \in \mathbb{R}^{2K} \qquad (9)$$

Then, the loss is computed as:

$$\mathcal{L}_s = 2 - 2 \cdot \frac{<\mathbf{s}, \mathbf{b}>}{\|\mathbf{s}\|_2 \cdot \|\mathbf{b}\|_2}. \qquad (10)$$

At each step in the test-time adaptation, we recompute $\mathcal{Z}^+$ and $\mathcal{Z}^-$. After $T$ steps, the adapted model assigns a final score to each frame in $\mathcal{V}$. We compute a moving average of these scores to further enhance temporal consistency. We then obtain temporal action proposals, $i.e.$, start/end time displacements $\{\hat{t}_i\}_{i=1}^{\hat{M}}$, by filtering with a threshold $\gamma$. Instead of using a fixed threshold, we set $\gamma$ as the average value of the scores along the whole video. After filtering, we group consecutive frames into region proposals and update each region label $\hat{y}_i$ with the class that maximizes the cosine similarity of the region-level representation, defined as the average of its frames.

### 4.4. Text-guided region suppression

The last step aims to reduce potential incorrectly predicted action proposals. To achieve this, we utilize the semantic

guidance of a existing captioning model, to contribute in identifying semantic variations from the textual modality.

First, all frames belonging to the selected action proposals are captioned. Then, we feed the obtained captions to the language encoder $\mathcal{E}_L$, averaging the textual representations obtained within each temporal proposal $\hat{t}_i$ to get a region-level representation $d_i$ that carries semantic information. We establish a rejection criteria by calculating the pairwise cosine similarity among all these representations. To this aim, we define the matrix of pairwise cosine similarities as:

$$\mathbf{D} = [d_{ij}], \quad d_{ij} = \cos(d_i, d_j) \tag{11}$$

Then, we binarize it at a threshold $\beta$ and obtain $\hat{\mathbf{D}}$. Summing up column-wise the elements in this binary mask, we obtain a score vector $\mathbf{d} = \hat{\mathbf{D}} \operatorname{diag}\left(\mathbf{I}_{\hat{M}}\right) \in \mathbb{R}^{\hat{M}}$ that measures the similarity of each action proposal with the others.

At last, a proposal $\hat{t}_i$ is suppressed if its associated entry in $\mathbf{d}$ is below a threshold $\alpha$, *i.e.*, its associated textual representation is insufficiently close to the others. As a result, we obtain $\{(y_i, t_i)\}_{i=1}^M$, with $M \leq \hat{M}$. After making a prediction on $\mathcal{V}$, the optimized parameters $\left(\theta_{\mathcal{P}_V}^*, \theta_{\mathcal{P}_L}^*, \tau^*\right)$ are discarded and re-initialized with the original ones.

We adopt the CoCa [31] model. This model follows a dual encoder architecture and includes an extra text decoder, and is trained with a contrastive loss and a captioning loss. This model is particularly convenient for our approach since it allows us to adopt a single model to perform action classification as well as proposal generation and suppression, via its the textual/visual encoder and the underlined captioner.

## 5. Experiments

**Datasets and settings.** We conduct experiments with two popular untrimmed video datasets, *i.e.*, ActivityNet-v1.3 [6] and THUMOS14 [7]. ActivityNet-v1.3 contains 19,994 videos describing 200 action classes, while THUMOS14 has 413 videos from 20 categories. Following [9], we validate our approach by dividing dataset classes into training and testing. We consider a 50%-50% split and a 75%-25% split. To guarantee statistical significance, we repeat class sampling ten times per split, reporting their average.
**Metrics.** We report the mean Average Precision (mAP) computed at different temporal IoU thresholds. Following prior work [9, 20], our tables include mAP at IoU thresholds of [0.3:0.1:0.7] for the THUMOS14 dataset and [0.5:0.05:0.95] for the ActivityNet-v1.3 dataset.
**Implementation details.** We extract RGB frames maintaining the original frame rate and resizing them to a resolution of $224 \times 224$. Class names are augmented with the prompt "a video of action {CLS}", for both $T3AL$ and the baselines defined in Sec. 5.1. We use CoCa (ViT-L/14) with the implementation of [8]. We adapt it for a maximum of $T = 50$ steps on THUMOS14 and $T = 25$ on ActivityNet-v1.3, with early stopping on the individual sample: if the loss

| Method | mAP (%) ↑ | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | Avg. |
| CLIP$_{32}$ [22] | 7.2 | 4.1 | 2.3 | 1.1 | 0.5 | 3.0 |
| CLIP$_{16}$ [22] | 7.5 | 4.2 | 2.2 | 1.1 | 0.6 | 3.1 |
| CoCa [31] | 8.4 | 4.7 | 2.5 | 1.2 | 0.6 | 3.5 |
| $T3AL_{T=0}$ | 11.4 | 6.8 | 3.5 | 1.7 | 0.6 | 4.8 |
| $T3AL$ | 20.7 | 14.3 | 8.9 | 5.3 | 2.7 | 10.4 |
| CLIP$_{16}$ w/ Detector [9, 20] | 27.2 | 21.3 | 15.3 | 9.7 | 4.8 | 15.7 |
| EffPrompt [9] | 37.2 | 29.6 | 21.6 | 14.0 | 7.2 | 21.9 |
| STALE [20] | 38.3 | 30.7 | 21.2 | 13.8 | 7.0 | 22.2 |

Table 1. **Results on THUMOS14 (50%-50%).** Green is our method , purple indicates training-based approaches.

| Method | mAP (%) ↑ | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | Avg. |
| CLIP$_{32}$ [22] | 5.5 | 3.3 | 1.9 | 0.9 | 0.4 | 2.4 |
| CLIP$_{16}$ [22] | 6.9 | 3.8 | 2.1 | 1.1 | 0.6 | 2.9 |
| CoCa [31] | 7.8 | 4.6 | 2.5 | 1.3 | 0.6 | 3.4 |
| $T3AL_{T=0}$ | 11.1 | 6.5 | 3.2 | 1.5 | 0.6 | 4.6 |
| $T3AL$ | 19.2 | 12.7 | 7.4 | 4.4 | 2.2 | 9.2 |
| CLIP$_{16}$ w/ Detector [9, 20] | 33.0 | 25.5 | 18.3 | 11.6 | 5.7 | 18.8 |
| EffPrompt [9] | 39.7 | 31.6 | 23.0 | 14.9 | 7.5 | 23.3 |
| STALE [20] | 40.5 | 32.3 | 23.5 | 15.3 | 7.6 | 23.8 |

Table 2. **Results on THUMOS14 (75%-25%).** Green is our method , purple indicates training-based approaches.

does not diminish after 5 consecutive steps, the adaptation process is halted and we proceed to the final prediction. We use Adam optimizer with a learning rate of $1e^{-5}$, and set $\alpha = 0.5$, $\beta = 0.75$, and $K = 4/20$ for THUMOS14 and ActivityNet-v1.3. We empirically observe that subtracting $y^*$ from the visual features improves the performance on THU-MOS14, augmenting the discrimination between features belonging to foreground and background of the action. We observe the opposite on ActivityNet-v1.3 and attribute this behaviour to the different average video length. We therefore remove the background information only for THUMOS14. All the experiments are conducted using one NVIDIA V100 GPU in floating point precision.

### 5.1. Comparative results

As we are unaware of methods designed for ZS-TAL that abstain from training on labeled data, we propose three baselines on top of pre-trained VLMs: CLIP (ViT-B/32), CLIP (ViT-B/16) [22], and CoCa (ViT-L/14) [31]. In the following, we will refer to these as CLIP$_{32}$, CLIP$_{16}$, and CoCa. For each of the three, the naive approach for TAL consists of independently classifying the video frames. We evaluate the cosine similarity between frame-level representations and textual descriptions generated via prompting on the class names. We convert their image-text cosine similarities into probabilities with the softmax operator. To recognize frames

| Method | mAP (%) ↑ | | | |
|---|---|---|---|---|
| | 0.50 | 0.75 | 0.95 | Avg. |
| CLIP$_{32}$ [22] | 0.4 | 0.2 | 0.0 | 0.2 |
| CLIP$_{16}$ [22] | 0.8 | 0.3 | 0.0 | 0.3 |
| CoCa [31] | 2.3 | 1.0 | 0.2 | 1.1 |
| $T3AL_{T=0}$ | 24.2 | 13.0 | 2.8 | 13.3 |
| $T3AL$ | 25.8 | 13.9 | 3.1 | 14.3 |
| CLIP$_{16}$ w/ Detector [9, 20] | 28.0 | 16.4 | 1.2 | 16.0 |
| EffPrompt [9] | 32.0 | 19.3 | 2.9 | 19.6 |
| STALE [20] | 32.1 | 20.7 | 5.9 | 20.5 |
| UnLoc [30] | 43.7 | - | - | - |

Table 3. **Results on ActivityNet-v1.3 (50%-50%).** Green is our method , purple indicates training-based approaches.

| Method | mAP (%) ↑ | | | |
|---|---|---|---|---|
| | 0.50 | 0.75 | 0.95 | Avg. |
| CLIP$_{32}$ [22] | 0.4 | 0.1 | 0.0 | 0.2 |
| CLIP$_{16}$ [22] | 0.9 | 0.3 | 0.1 | 0.4 |
| CoCa [31] | 3.1 | 1.3 | 0.3 | 1.6 |
| $T3AL_{T=0}$ | 26.1 | 13.9 | 2.9 | 14.3 |
| $T3AL$ | 28.1 | 14.9 | 3.3 | 15.4 |
| CLIP$_{16}$ w/ Detector [9, 20] | 35.6 | 20.4 | 2.1 | 20.2 |
| EffPrompt [9] | 37.6 | 22.9 | 3.8 | 23.1 |
| STALE [20] | 38.2 | 25.2 | 6.0 | 24.9 |
| UnLoc [30] | 48.8 | - | - | - |

Table 4. **Results on ActivityNet-v1.3 (75%-25%).** Green is our method , purple indicates training-based approaches.

as actions or background, we binarize their probabilities for the predicted class. For this, we apply a threshold of $0.8$. Following previous work [9, 20], we report a two-stage baseline (which we call CLIP$_{16}$ w/ Detector) consisting of a pre-trained proposal detector [14] and CLIP as the second-stage proposal classifier. As an additional baseline, we present our method with $T = 0$ steps of adaptation, denoted as $T3AL_{T=0}$. Tab. 1 and Tab. 2 show results on THUMOS14 for the 50%-seen 50%-unseen and 75%-seen 25%-unseen splits; Tab. 3 and Tab. 4 present the same class splits on ActivityNet-v1.3. For the ease of readers, we also report results on state-of-the-art models achieved through training. [9, 20, 30]. All the tables suggest that a naive application of VLMs is insufficient for the ZS-TAL task. We further show that a simple use of the video-level pseudo-labelling, as detailed in Sec. 4, can considerably improve results of these baselines. $T3AL_{T=0}$ without TTA achieves an average improvement of +1.2% mAP on THUMOS14, and of +12.4% mAP on ActivityNet-v1.3. With test-time learning we improve further, with an extra gain of +1.0% and +5.1% mAP on ActivityNet-v1.3 and THUMOS14. Refer to the *Supplementary Material* for additional results.

| Configuration | | | mAP (%) ↑ | | | | | |
|---|---|---|---|---|---|---|---|---|
| Loss | $\theta_{\mathcal{P}_V}$ | $\theta_{\mathcal{P}_L}$ | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | Avg. |
| BCE | ✓ | | 16.7 | 10.8 | 6.4 | 3.5 | 1.8 | 7.9 |
| $\mathcal{L}_s$ | | ✓ | 18.1 | 11.7 | 6.9 | 3.7 | 1.7 | 8.4 |
| $\mathcal{L}_z$ | | ✓ | 20.9 | 14.3 | 8.7 | 4.9 | 2.5 | 10.3 |
| $\mathcal{L}_z + \mathcal{L}_s$ | ✓ | ✓ | 20.7 | 14.3 | 8.9 | 5.3 | 2.8 | 10.4 |

Table 5. **Ablation on learning objective.** Green is our configuration . Results are collected on THUMOS14 (50%-50%).

| Method | Split | mAP (%) ↑ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | Avg. |
| $T3AL_{no\_sup}$ | 50:50 | 20.1 | 13.9 | 8.7 | 5.2 | 2.6 | 10.1 |
| $T3AL$ | 50:50 | 20.7 | 14.3 | 8.9 | 5.3 | 2.8 | 10.4 |
| $T3AL_{no\_sup}$ | 75:25 | 18.6 | 12.2 | 7.0 | 4.2 | 2.1 | 8.8 |
| $T3AL$ | 75:25 | 19.2 | 12.7 | 7.4 | 4.4 | 2.2 | 9.2 |

Table 6. **Ablation on text-guided region suppression.** Green is our configuration . Results are collected on THUMOS14 (50%-50%) and on THUMOS14 (75%-25%).

## 5.2. Ablation

We thoroughly ablate $T3AL$ on the THUMOS14 dataset to validate the effectiveness of our design choices. First, we analyze the learning objective functions, followed by the selected fine-tuned parameters and the final suppression step. Furthermore, we conduct oracle experiments to showcase the potential of our approach in addressing this challenging scenario. When not state otherwise, we report results for the 50%-seen 50%-unseen split, averaged across the 10 splits to guarantee statistical significance.

**Learning objective.** In Tab. 5 we analyze the learning objective and the selected parameters for adaptation. We compare different configurations of the loss defined in Sec. 4.3 and a binary cross entropy (BCE) loss on the same input of $\mathcal{L}_z$. Notably, incorporating the loss on the representations (see Row 3-4) improves performance compared to solely utilizing the loss on the scores (see Row 1-2). When the Representation loss is added to the Separation loss and we adapt both the vision and language projection layers, we observe an improvement up to +2.0% mAP.

**Text-guided suppression.** We assess the efficacy of the final text-guided suppression (Sec. 4.4) in Tab. 6, and report results for the 75:25 and 50:50 splits. The table include results for when we apply this suppression (see Row 2-4) and when we do not (see Row 1-3). Our findings indicate a positive contribution from the suppression in all the settings.

**Oracle analysis** To validate the potential of our proposed methodology, we conduct an extensive study on top of $T3AL$. These analyses investigate the potential of our TTA method under the relaxation of certain unsupervised constraints. Starting from our method design, we identify three components we can replace with oracle information: perfect
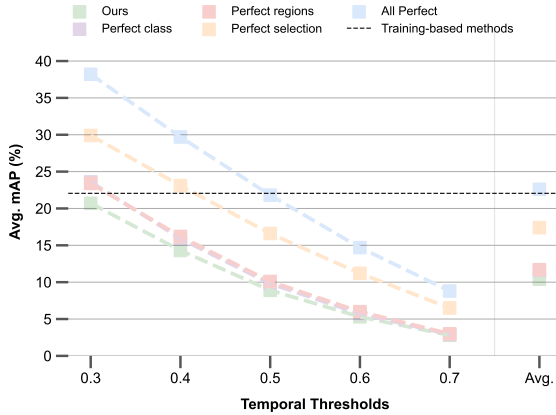
Figure 4. **Oracle study.** We re-evaluate our configuration with partial perfect information as perfect class prediction for the pseudo-label, perfect regions count selection in the video, and perfect selection of positive and negative refinement samples. With all perfect mechanisms, we surpass training-based models.

video-level pseudo-label, perfect region count, and perfect positive selection. We report the performance fluctuation derived from oracle knowledge in Fig. 4.

In the first experiment, we account for an imaginary classifier able to recognize with 100% accuracy the action from the average representation of the video frames. We replace the proposed pseudo-label $y^*$ with such a perfect prediction. As shown in Fig. 4, a better classifier achieves up to +1.2% mAP. In the second setting, we re-evaluate the performance of our suppression strategy when we select the exact number of action regions $m$ in the video. In this case, after adaptation, we rank the predicted region proposals based on their similarity with the pseudo-label and retain only the first $m$. Similar to the previous analysis, perfect region count results in a relative gain of +1.3% mAP. Last, we consider the scenario where we retrieve positive samples from intervals encapsulating the video-level pseudo-label and retrieve negative samples from outside such intervals. These results capitalize on a considerable gain in performance, surpassing the improvements on the previous two, with a final score of 17.4%, *i.e.*, +7% relative gain. The final experiment consider all the aforementioned constraints relaxation at once. In this configuration, the model performance increase further and achieves 22.6% on average. Remarkably, these numbers are on on par with state-of-the-art models models evaluated in-domain, without requiring training data or human annotations. To maintain comparability, the maximum number of adaptation steps used in the main method is kept consistent across all oracle experiments, even if there could potentially be further improvements to the oracle.

## 6. Discussion

Our research highlighted a shortcoming in existing ZS-TAL literature, indicating an inability of such models for out-of-distribution generalization. Motivated by this observation, we propose $T3AL$, a novel approach based on test-time adaptation to fine-tune the model without any training data. Our method expands a generic VLM, *i.e.*, pre-trained on image data without fine-tuning for TAL, to jointly adapt to video data and learn to localize actions in a zero-shot manner. We achieve all this by only adapting on unlabelled video samples individually. Our experimental evaluation confirms test-time adaptation as a promising direction to 1) calibrate VLMs to solve action localization in videos and 2) mitigate the out-of-distribution generalization problem of current ZS-TAL approaches. Moreover, our study on partial perfect information reveals that test-time adaptation can achieve and surpass current state-of-the-art implementations without training on labeled samples.

**Limitations.** $T3AL$ relies heavily on good positive and negative samples, which are essential for adequate adaptation to unlabelled data. Our selection protocol tags the frames semantically closer to the video pseudo-label as positives and the ones that are less similar as negatives. Negatives selected in this way, however, may contain completely unrelated concepts such as titles or black screens. Such samples are suboptimal compared to more informative hard negatives. These hard negatives, such as frames that are highly correlated with the actions in videos that are not part of the ground truth regions, provide superior information for the adaptation process. Additionally, the video-level pseudo-label restricts to one the number of actions per video, idealizing the real-world setup where multiple actions may appear concurrently.

**Potential directions.** While we propose test-time adaptation to address the out-of-distribution problem of current ZS-TAL models, we acknowledge other directions as viable alternatives, such as cross-domain evaluation protocols (currently studied for video action recognition) or source-free approaches [29, 32]. We also believe that VLMs pre-trained for video data will provide a better starting point for temporal visual tasks, similar to a pre-trained action localizer, to adapt at test-time. Last, our results also partially highlight that annotated training data is not fundamental to outperform current state-of-the-art ZS-TAL methods. While not explored in this work, we hypothesize that such behavior might be associated with the inherent noise in the label space of action datasets. The existing annotation lacks a well-defined taxonomy, *i.e.*, current approaches must account for a mixture of action verbs, nouns describing actions, and activities (*i.e.*, succession of atomic actions). We advocate for a systematic action taxonomy as a vital future step towards better action-related vision tasks.

# References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikołaj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for Few-Shot learning. *NeurIPS*, 2022. 1

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

[3] 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

[4] Sachin Goyal, Ananya Kumar, Sankalp Garg, Zico Kolter, and Aditi Raghunathan. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *CVPR*, 2023. 1

[5] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent a new approach to self-supervised learning. In *NeurIPS*, 2020. 2, 5

[6] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 2, 6

[7] 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". *CVIU*, 2017. 2, 6

[8] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. 6

[9] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. In *ECCV*, 2021. 1, 2, 3, 6, 7

[10] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: A large video database for human motion recognition. In *ICCV*, 2011. 3

[11] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 1

[12] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping Language-Image pre-training with frozen image encoders and large language models. *arXiv*, 2023. 1

[13] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *CVPR*, 2022. 1

[14] Chuming Lin, Chengming Xu, Donghao Luo, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yanwei Fu. Learning salient boundary feature for anchor-free temporal action localization. In *CVPR*, 2021. 2, 3, 7

[15] Tianwei Lin, Xu Zhao, and Zheng Shou. Single shot temporal action detection. In *ACMMM*, 2017. 2

[16] Wei Lin, Muhammad Jehanzeb Mirza, Mateusz Kozinski, Horst Possegger, Hilde Kuehne, and Horst Bischof. Video test-time adaptation for action recognition. In *CVPR*, 2023. 3

[17] Xiaosong Ma, Jie Zhang, Song Guo, and Wenchao Xu. Swap-prompt: Test-time prompt adaptation for vision-language models. In *NeurIPS*, 2023. 3

[18] Shu Manli, Nie Weili, Huang De-An, Yu Zhiding, Goldstein Tom, Anandkumar Anima, and Xiao Chaowei. Test-time prompt tuning for zero-shot generalization in vision-language models. In *NeurIPS*, 2022. 3

[19] Liliane Momeni, Mathilde Caron, Arsha Nagrani, Andrew Zisserman, and Cordelia Schmid. Verbs in action: Improving verb understanding in video-language models. In *ICCV*, 2023. 2

[20] Sauradip Nag, Xiatian Zhu, Yi-Zhe Song, and Tao Xiang. Zero-shot temporal action detection via vision-language prompting. In *ECCV*, 2022. 1, 2, 3, 6, 7

[21] Mirco Plananamente, Chiara Plizzari, and Barbara Caputo. Test-time adaptation for egocentric action recognition. In *ICIAP 2022*, 2022. 3

[22] 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, 2, 6, 7

[23] Jameel Hassan Abdul Samadh, Hanan Gani, Noor Hazim Hussein, Muhammad Uzair Khattak, Muzammal Naseer, Fahad Khan, and Salman Khan. Align your prompts: Test-time prompting with distribution alignment for zero-shot generalization. In *NeurIPS*, 2023. 3

[24] Yu Sun, Xiaolong Wang, Liu Zhuang, John Miller, Moritz Hardt, and Alexei A. Efros. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, 2020. 2

[25] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021. 2

[26] Wenhao Wu, Xiaohan Wang, Haipeng Luo, Jingdong Wang, Yi Yang, and Wanli Ouyang. Bidirectional cross-modal knowledge exploration for video recognition with pre-trained vision-language models. In *CVPR*, 2023. 1

[27] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *ICCV*, 2017. 2

[28] 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. *arXiv*, 2021. 1

[29] Yuecong Xu, Jianfei Yang, Haozhi Cao, Keyu Wu, Min Wu, and Zhenghua Chen. Source-free video domain adaptation by learning temporal consistency for action recognition. In *ECCV*, 2022. 8

[30] Shen Yan, Xuehan Xiong, Arsha Nagrani, Anurag Arnab, Zhonghao Wang, Weina Ge, David Ross, and Cordelia Schmid. Unloc: A unified framework for video localization tasks. In *ICCV*, 2023. 1, 2, 3, 7

[31] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv*, 2022. 1, 4, 6, 7

[32] Giacomo Zara, Alessandro Conti, Subhankar Roy, Stéphane Lathuilière, Paolo Rota, and Elisa Ricci. The unreasonable effectiveness of large language-vision models for source-free video domain adaptation. In *ICCV*, 2023. 1, 8

[33] Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. In *NeurIPS*, 2022. 2

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

[35] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, 2022. 1