

The Blessings of Unlabeled Background in Untrimmed Videos

Yuan Liu¹ Jingyuan Chen¹ Zhenfang Chen²
Bing Deng¹ Jianqiang Huang¹ Hanwang Zhang³

¹Alibaba Group ²The University of Hong Kong ³Nanyang Technological University

lhy19930911@gmail.com jingyuanchen91@gmail.com zfchen@cs.hku.hk
dengbing.db@alibaba-inc.com jianqiang.jqh@gmail.com hanwangzhang@ntu.edu.sg

Abstract

Weakly-supervised Temporal Action Localization (WTAL) aims to detect the action segments with only video-level action labels in training. The key challenge is how to distinguish the action of interest segments from the background, which is unlabelled even on the video-level. While previous works treat the background as “curses”, we consider it as “blessings”. Specifically, we first use causal analysis to point out that the common localization errors are due to the unobserved confounder that resides ubiquitously in visual recognition. Then, we propose a Temporal Smoothing PCA-based (TS-PCA) deconfounder, which exploits the unlabelled background to model an observed substitute for the unobserved confounder, to remove the confounding effect. Note that the proposed deconfounder is model-agnostic and non-intrusive, and hence can be applied in any WTAL method without model re-designs. Through extensive experiments on four state-of-the-art WTAL methods, we show that the deconfounder can improve all of them on the public datasets: THUMOS-14 and ActivityNet-1.3¹.

1. Introduction

Temporal action localization aims to locate the start and end frames of actions of interest in untrimmed videos, *e.g.*, CLIFFDIVE and PASSBALL. Weakly-supervised Temporal Action Localization (WTAL) [21, 35, 37] can train such a localizer by using only the video-level annotations like “the video contains CLIFFDIVE”, without specifying its start and end frames. Therefore, WTAL is especially valuable in the era of video-sharing social networking service [9], where billions of videos have only video-level user-generated tags.

Not surprisingly, WTAL has lower performances than its fully-supervised counterpart [8, 15, 17]. As shown in Figure 1, any WTAL method would suffer from the three

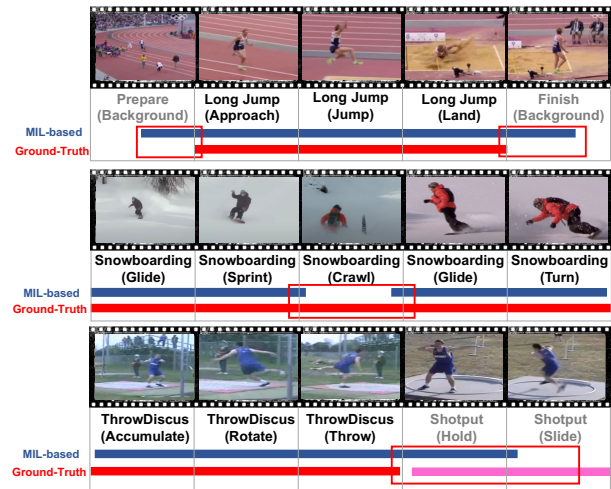


Figure 1: Three types of common errors in existing MIL-based WTAL models. Top to Bottom: over-completeness, Incompleteness, Confusion. We argue that the errors are due to the unobserved confounders that affects the video (including both foreground and background) and the label simultaneously.

types of localization errors due to the weak supervision: 1) *Over-completeness*: the ground-truth is a sub-sequence of the localization, which contains additional background frames (Top). 2) *Incompleteness*: the localization only selects the discriminative sub-sequence of the ground-truth, *i.e.*, misclassifying foreground into background (Mid). 3) *Class Confusion*: the localization misclassifies different actions (Bottom). Prior studies would always blame these errors for the unlabeled background segments, because the reasons are intuitively sound. For example, if we could model the background as a special BACKGROUND class, error 1 and error 2 would be mitigated since WTAL knows what visual cues are background [13, 16]; if we could exploit the difference between the background and foreground, each action class model would be more discriminative, thanks to the training on more negative background

¹Code is available at <https://github.com/liuyuancv/WTAL.blessing>

samples [14, 21, 24]—all the three errors are addressed.

However, it is impossible to fundamentally resolve the “background” issue by only using video-level supervision. The reasons are two-fold. First, as there is neither foreground nor background video-level labels, even if we set up a BACKGROUND class, the BACKGROUND model will be eventually reduced to a simple prior, such as the foreground is sparsely distributed in the background [21, 33]. Second, learning more discriminative features between background and foreground is no easier than WTAL *per se*, because if we have had such features, videos would be trimmed into segments of foreground actions, which are no longer weakly-supervised anymore. As a result, all such methods essentially resort to bootstrapping from another trivial prior [20, 24]: segments of the same action should look similar, which is however not always the case for complex actions (*e.g.*, THROWDISCUS in Figure 1).

In this paper, we propose a *causal inference* framework [25, 31] to identify the true cause—which is indeed not the background—of the WTAL deficiency, and then how to remedy it by eliminating the cause. To begin with, we help you to find out the possible confounders in Figure 1. In causal terms, a *confounder* is the common cause of many resultant effects, which are observed to be correlated even if they share no causation. For example, the context “athletic track” may be the confounder between the video and its video-level label LONGJUMP—even if the “Prepare”/“Finish” background is not the action, it is inevitably correlated with the foreground and the label, misleading WTAL. The contextual confounder for the THROWDISCUS and SHOTPUT confusion can be found similarly. As another example, the object “leg on snowboard” may be the confounder for the SNOWBOARDING video, where if a segment has no such object, it is wrongly detected as negative.

It is well-known that such confounding bias can be *never* removed by using only the observational data correlation [26], *e.g.*, the action likelihood estimated from a Multiple Instance Learning (MIL) model [20]. Instead, we should perform *causal intervention* to remove the confounding effect. Ideally, if we can observe all the possible confounders, we can estimate the average effect of the video evenly associated with each confounder [27, 45], mimicking the physical intervention such as “recording” the videos of LONGJUMP everywhere. Unfortunately, the above observed-confounder assumption is not valid in general, because the confounder is too elusive to be defined in different actions that have complex visual cues, *e.g.*, it can be the scene context in LONGJUMP or the object in SNOWBOARDING. Fortunately, we can use another way around: Deconfounder [38], which is a general theory that uses latent variables, who can generate the observed data, to be a good substitute for the confounders.

To implement the deconfounder for WTAL, we pro-

pose an unsupervised Temporal Smoothing PCA (TS-PCA) model whose base can reconstruct the whole video dataset, whose majority consists of the unlabelled background segments (Section 3.2.2). Therefore, our punchline is that the background is indeed not a curse but blessings for WTAL. It is worth noting that the TS-PCA deconfounder is specially designed that each projection directly contributes to the label prediction logits. This has two crucial benefits: 1) Its implementation is model-agnostic, *i.e.*, any existing WTAL model can seamlessly incorporate it in a plug-and-play fashion (Section 3.2.1). 2) The TS-PCA projection can be used as a foreground/background score function that further enhances the prediction (Section 3.2.2). The deconfounder model is applied to four state-of-the-art WTAL methods with public codes on THUMOS-14 and ActivityNet-1.3 datasets, where a significant improvement is observed and the deconfounded WUM [14] establishes a new state-of-the-art performance (Section 4).

2. Related Work

Weakly-supervised Temporal Action Localization [10, 14, 20, 24]. UntrimmedNets [37] locates action instances by selecting relevant segments in soft and hard ways. Autoloc [35] directly predicts the temporal boundary of each action instance with an outer-inner-contrastive loss. BasNet [13] proposes an auxiliary class representing background to suppress the activation from background frames. However, these methods have localization errors: over-completeness and incompleteness. Many approaches have been proposed to tackle the issues by erasing salient features [36, 47], imposing diversity loss [16], and employing marginalized average aggregation strategy [41]. Nevertheless, these methods ignore the challenging action-context confusion issue caused by the absence of frame-wise label. DGAM [33] builds a generative model for the frame representation, helpful for the action-context separation. However, all components of DGAM are coupled together, which cannot be utilized in a plug-and-play fashion. In this paper, we propose an unsupervised TS-PCA deconfounder to solve the WTAL problem by contributing a principled answer based on causal inference.

Causal Inference. Causal inference can be used to remove the spurious bias [2] and disentangle the desired model effects [3] in domain-specific applications by pursuing the causal effect [28, 32]. Recently, causal inference has been introduced and widely used in various computer vision tasks [23, 29, 45], including image classification [7], few-shot learning [42], and semantic segmentation [45]. In our work, as the confounder is unobserved, those deconfounding methods based on the observed-confounder assumption is no longer valid. To this end, we propose to derive a substitute confounder from a fitted data generation model [38].

3. Approach

In this section, we discuss how to derive our deconfounding techniques from the causal effect perspective and how to implement it into the prevailing WTAL framework.

3.1. Action Localization as Causal Effects

Task Definition. Weakly-supervised temporal action localization (WTAL) aims to predict a set of action instances given an untrimmed video. Suppose we have N training videos $\{\mathbf{x}_i\}_{i=1}^N$ with their video-level labels $\{\mathbf{y}_i\}_{i=1}^N$, where \mathbf{y}_i is a F -dimensional multi-hot vector representing the presence of F action categories and $\mathbf{x}_i \in \mathbb{R}^{T \times D}$ is the extracted D -dimensional video features of T segments.

The prevailing pipeline for training WTAL is illustrated in the blue dashed box of Figure 2. Note the whole framework is trained with only video-level action labels. First, video \mathbf{x}_i goes through an action classifier to generate segment-wise scores for each action category, namely, Class Activation Sequences (CASs), which are denoted as $A(\mathbf{x}_i) \in \mathbb{R}^{T \times F}$. The CAS score reflects the probability of each segment whether it belongs to a specific action category. Then, CASs are aggregated to produce the video-level class scores $\mathbf{y}_i = \varphi(A(\mathbf{x}_i)) \in \mathbb{R}^F$. Specifically, $\varphi(\cdot)$ is usually a rule-based function and the most widely used one is top- k mean method [13, 20, 24], which averages the top- k scores of all segments for each action category, followed by a softmax operation to derive the video-level action probabilities. Therefore, the video-level classification model can be seen as a proxy objective for CAS learning, with a binary cross-entropy loss for each class as:

$$\mathcal{L}_{cls} = -\frac{1}{N} \sum_{i=1}^N \sum_{f=1}^F y_{i,f} \log(\varphi(A(\mathbf{x}_i))_f). \quad (1)$$

During inference, action instances are localized by thresholding $A(\mathbf{x}_i)$ for segment-wise action intervals. However, as we discussed in Section 1, it is essentially ill-posed to infer $A(\mathbf{x}_i)$ from only video-level labels via Eq. (1) due to the unobserved confounder. In the following discussion, when the context is clear, we omit the subscript of each sample for simplicity.

Causal Effect Analysis. We slightly abuse the notation A to denote that there exists an “ideal” CAS A for the ground-truth video-level action label \mathbf{y} . In causal inference, A is also known as the *potential outcome* [31], and the challenge is how to use and only use the training data to estimate it. Denote the conditional Monte Carlo approximations [4] of A as $\mathbb{E}[A|X = \mathbf{x}]$. However, as shown in Figure 3, when there is confounder C , it will affect both the input video features \mathbf{x} and CAS A , i.e., \mathbf{x} may correlate with the observed label \mathbf{y} , even if \mathbf{x} is not the true cause (Recall the localization errors illustrated in Figure 1). Therefore, the Monte

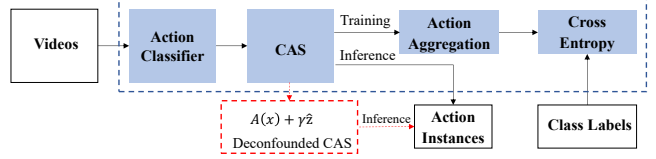


Figure 2: The blue dashed box denotes the prevailing pipeline for WTAL. Our contribution is to calibrate CAS with the deconfounder during the inference process, which is shown in the red dashed box.

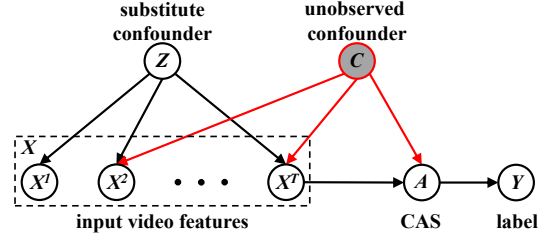


Figure 3: The unobserved confounders C are correlated to the input video X and CAS A . The substitute confounder Z captures the distribution of all the input video features. Therefore, all the confounders are captured with the substitute confounders.

Carlo estimation $\mathbb{E}[A|X = \mathbf{x}]$ is biased,

$$\mathbb{E}[A|X = \mathbf{x}] \neq A. \quad (2)$$

Suppose we could observe and measure all the possible confounders C and associate them to each data sample,

$$\mathbb{E}_c [\mathbb{E}[A|X = \mathbf{x}, C = c]] = A, \quad (3)$$

which is also known as the backdoor adjustment [26]. Unfortunately, the above assumption of observed confounder is not valid in general because the confounder is ever-elusive to be defined in different actions of complex visual cues. Next, we propose a deconfounder method to remove the confounding effect.

3.2. Deconfounded WTAL

We now develop the deconfounder [38] for WTAL task, which is a general algorithm that uses latent variables which generate the observed data, as a good substitute for the unobserved confounders (Section 3.2.1). Note that the substitute is obtained from an unsupervised generative model learned from all videos—regardless of foreground or background, labeled or unlabeled—the blessings of unlabeled background (Section 3.2.2).

3.2.1 Deconfounded CAS Function

The deconfounder theory infers a substitute confounder to perform causal inference. As shown in Figure 3, we assume that there exists a substitute confounder Z which

could generate the distribution of the input video features $\mathbf{x} = \{\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^T\}$. Therefore, all the segment-level features are conditionally independent given \mathbf{z} :

$$P(\mathbf{x}^1, \dots, \mathbf{x}^T \mid Z = \mathbf{z}) = \prod_{t=1}^T P(\mathbf{x}^t \mid Z = \mathbf{z}). \quad (4)$$

Now we show that \mathbf{z} is a good substitute which can capture *all* the unobserved confounders C by contradiction. Suppose that there exists an unobserved confounder c , which affects multiple input video features within \mathbf{x} and segment-level labels A . Then, \mathbf{x} would be dependent, even conditional on \mathbf{z} , due to the impact of c . This dependence leads to a contradiction that Eq. (4) does not hold.

Given the substitute confounder \mathbf{z} who can generate \mathbf{x} , we can replace the expectation over C with a single \mathbf{z} in Eq. (3), thanks to the (weak) ignorability [11, 30]:

$$A = \mathbb{E}[A \mid X = \mathbf{x}, Z = \mathbf{z}] := \mathbb{E}[G(\mathbf{x}, \mathbf{z})], \quad (5)$$

where $G(\cdot)$ is the outcome model to estimate A . In particular, if we denote $\hat{\mathbf{z}}$ as a deterministic inference result (*i.e.*, decoded) from the generative model $\hat{\mathbf{z}} \sim P(\mathbf{z} \mid \mathbf{x})$, the expectation in Eq. (5) will be collapsed to:

$$A = \mathbb{E}[G(\mathbf{x}, \hat{\mathbf{z}})] = G(\mathbf{x}, \hat{\mathbf{z}}). \quad (6)$$

We propose to use a decoupled addition model to be the overall deconfounded CAS function $\hat{A}(\mathbf{x})$:

$$\hat{A}(\mathbf{x}) := G(\mathbf{x}, \hat{\mathbf{z}}) = A(\mathbf{x}) + \gamma \hat{\mathbf{z}}, \quad (7)$$

where $A(\cdot)$ is the conventional CAS function in any WTAL model, and γ is a trade-off parameter. It is worth noting that the deconfounded CAS $\hat{A}(\mathbf{x})$ is model-agnostic—all we need to achieve deconfounded WTAL is just to calibrate any trained and conventional $A(\mathbf{x})$ by adding $\gamma \hat{\mathbf{z}}$ as shown in the dashed red box of Figure 2. Before we detail how to obtain $\hat{\mathbf{z}}$ in Section 3.2.2, we first outline the deconfounded WTAL based on the deconfounded CAS function in Eq. (7).

Algorithm 1 illustrates the overview of the deconfounded WTAL. The inputs are training videos with only video-level class labels and the output is the deconfounded CAS scores $\hat{A}(\mathbf{x})$. Given any WTAL model $A(\mathbf{x})$ and the generative model which is to encode $\hat{\mathbf{z}}$, the two models are trained separately. With such decoupled training scheme, any existing WTAL model can be seamlessly incorporated with $\hat{\mathbf{z}}$ in a plug-and-play fashion. Specifically, $A(\mathbf{x})$ is typically trained with the loss function in Eq. (1) and the generative model is trained with \mathcal{L}_{PCA} detailed in Eq. (8). During the inference process, based on the trained $A(\mathbf{x})$ and the generated $\hat{\mathbf{z}}$, Eq. (7) is adopted to generate the deconfounded CAS scores, which is also shown in Figure 2. Action instances are inferred by thresholding $\hat{A}(\mathbf{x}_i)$, which is implemented with existing methods [13, 20, 24].

Algorithm 1: WTAL with the Deconfounder

Input: input videos and action categories $\{\mathbf{x}, \mathbf{y}\}$.

Output: Deconfounded CAS $\hat{A}(\mathbf{x})$ for each \mathbf{x} .

- 1: **repeat**
 - 2: Training any WTAL model with \mathcal{L}_{cls} in Eq. (1).
 - 3: Training TS-PCA with \mathcal{L}_{PCA} in Eq. (8).
 - 4: **until** Converge.
 - 5: **for** each video \mathbf{x} **do**
 - 6: Generating $A(\mathbf{x})$ with the trained WTAL model.
 - 7: Generating $\hat{\mathbf{z}}$ by projecting \mathbf{x} with the trained projectors of TS-PCA.
 - 8: Generating final result $\hat{A}(\mathbf{x})$ according to Eq. (7).
 - 9: **end for**
-

3.2.2 Temporal Smoothing PCA (TS-PCA)

To be a good substitute confounder, $\hat{\mathbf{z}}$ needs to satisfy: 1) it captures the joint distribution of input features \mathbf{x} ; and 2) $G(\mathbf{x}, \hat{\mathbf{z}})$ is model-agnostic, which requires $\hat{\mathbf{z}}$ to generate CAS scores to be seamlessly fused with any WTAL model. Subject to these constraints, we propose to use a temporal smoothing PCA (TS-PCA) as a simple yet effective generative model for encoding the substitute confounder $\hat{\mathbf{z}}$. The model directly projects the features into a hidden factor $\hat{\mathbf{z}}$, which can be discriminative and naturally reflects the background or foreground. Without such unsupervised technique, it will require to train the projection together with $A(\mathbf{x})$, which ruins the decoupled training requirement of deconfounder.

Generally, TS-PCA learns L feature projectors $\mathbf{P} = \{\mathbf{p}_l\}_{l=1}^L$ to maximize the variance of the projected features \mathbf{x} . Given the training video set $\{\mathbf{x}\}$, the overall objective function of TS-PCA is:

$$\mathcal{L}_{PCA} = \mathcal{L}_{pro} + \lambda \mathcal{L}_{recon} + \beta \mathcal{L}_{smooth}, \quad (8)$$

where λ and β are parameters to control the relative contributions. \mathcal{L}_{pro} denotes the projection loss, which is the basis of PCA and composed of a variance and an orthogonalization term as:

$$\begin{aligned} \mathcal{L}_{pro} = & - \sum_t \sum_l \left(\mathbf{p}_l^T \mathbf{x}^t - \frac{\sum_{m=1}^T \mathbf{p}_l^T \mathbf{x}^m}{T} \right)^2 \\ & + \frac{1}{L^2} \|PP^T - I\|_F. \end{aligned} \quad (9)$$

Moreover, with the feature projectors, we can approximately reconstruct the original features by mapping it back with \mathbf{p}_l . The reconstruction loss \mathcal{L}_{recon} is as follows:

$$\mathcal{L}_{recon} = \sum_t \left\| \sum_l (\mathbf{p}_l^T \mathbf{x}^t) \mathbf{p}_l - \mathbf{x}^t \right\|^2. \quad (10)$$

Motivated by the fact that nearby segments tend to have similar visual semantics, we propose a temporal smoothing loss to penalize the feature projection differences of consecutive segments:

$$\mathcal{L}_{smooth} = \sum_t \sum_l (\mathbf{p}_l^\top \mathbf{x}^{t+1} - \mathbf{p}_l^\top \mathbf{x}^t)^2. \quad (11)$$

For each video, we can calculate the substitute confounder $\hat{\mathbf{z}}$ using the TS-PCA above:

$$\hat{\mathbf{z}} = \frac{1}{L} \sum_{l=1}^L \mathbf{p}_l^\top \mathbf{x}, \quad (12)$$

where $\hat{\mathbf{z}} \in \mathbb{R}^T$ is computed by projecting features with the trained projectors. We broadcast $\hat{\mathbf{z}}$ to $\mathbb{R}^{T \times F}$, which could be added to $\hat{\mathbf{A}}(\mathbf{x})$ shown in Eq. (7).

4. Experiments

Following previous works [13, 21], we conduct experiments on THUMOS-14 and ActivityNet-1.3 benchmarks. We show the effectiveness of the TS-PCA deconfounder on different baselines in Section 4.2. We then compare the proposed method with the state-of-the-art models in Section 4.3. We also validate the components in TS-PCA with detailed ablation studies in Section 4.4. Moreover, we show the false positive analysis and the qualitative results in Section 4.5 and Section 4.6, respectively.

4.1. Settings

Datasets. We evaluate the proposed deconfounder strategy on two standard datasets for WTAL, namely THUMOS-14 [12] and ActivityNet-1.3 [5]. **THUMOS-14** contains 200 videos and 212 videos with 20 action classes in the validation and test sets, respectively. Each untrimmed video contains at least one action category. Following the standard practice of previous works [13, 21], we train the network on the validation set and evaluate the trained model on the test set. **ActivityNet-1.3** is a large-scale video benchmark for temporal action localization, which consists of 19,994 videos with 200 classes annotated, with 50% for training, 25% for validation, and the rest 25% for testing. As in literature [24, 41], we train our model on the training set and perform evaluations on the validation set.

Evaluation Metrics. Following the standard evaluation metrics, we report mean average precision (mAP) at several different levels of intersection over union (IoU) thresholds. We use the official benchmarking code provided by [5] to evaluate our model on THUMOS-14 and ActivityNet-1.3.

Baseline Models. To demonstrate the generalizability of the proposed deconfounder strategy, we integrated it on four popular WTAL models including STPN [21], A2CL-PT [19], BasNet [13] and WUM [14] with available source

code. The four models utilize different strategies to improve the performance, serving as good baselines for demonstrating TS-PCA’s generalizability for different WTAL models. For a fair comparison, we follow the same settings as reported in the official codes of STPN, A2CL-PT, BasNet and WUM.

Implementation Details. I3D networks [6] are used to extract segment features which take segments with 16 frames as input. I3D networks are pre-trained on Kinetics [6]. Note that the feature extractors are not finetuned during training.

4.2. Effectiveness on Different Baselines

To demonstrate the generalizability of TS-PCA, we deploy it with four WTAL models on the THUMOS-14 dataset, which are STPN [21], A2CL-PT [19], BasNet [13] and WUM [14]. We also perform similar experiments with STPN and A2CL-PT [19] on the ActivityNet-1.3 dataset. Experimental results on THUMOS-14 are shown in Table 1, where “+TP” denotes that the TS-PCA deconfounder is deployed. We can observe that all four models with +TP obtain obvious improvements. Specifically, there are averaged mAP improvement of 1.18% on STPN, 0.99% on A2CL-PT, 1.01% on BasNet and 1.52% on WUM. As for ActivityNet-1.3 dataset, STPN and A2CL-PT with +TP also show similar performance improvement as shown in Table 2. Specifically, STPN+TP achieves an average mAP improvement of 1.8%, and A2CL-PT has an average improvement of 1.2%. These results show the effectiveness of TS-PCA deconfounder in calibrating CAS scores. Note that WUM is a recently proposed **SOTA** method, where the absolute improvements of 1.52% on THUMOS-14 are significant. The above results demonstrate the effectiveness and generalizability of the deconfounder.

4.3. Comparisons with State-of-the-Art Methods

We adopt the TS-PCA deconfounder on WUM as our model to compare with other methods on the testing set of THUMOS-14. We first compare WUM+TP with several baseline models in both weakly-supervised and fully-supervised training manners. The results are shown in Table 3. It can be observed that WUM+TP outperforms other weakly-supervised methods by a large margin, establishing new state-of-the-art performances.

Specifically, WUM+TP achieves an improvement of 5.3% on mAP at IoU=0.3 and 5.5% at IoU=0.6 compared with the second best model A2CL-PT. These results further demonstrate the effectiveness of the TS-PCA deconfounder in calibrating CAS. Moreover, our method can even achieve better results than some fully-supervised methods such as SSN [46] which is trained with temporal boundary annotations, showing the superiority of the proposed method.

Table 4 shows the comparison results on ActivityNet-1.3, where we adopt the TS-PCA deconfounder on A2CL-

Table 1: The deconfounder strategy is applied to different state-of-the-art WTAL models on THUMOS-14 dataset. “+TP” denotes the TS-PCA deconfounder strategy is deployed. mAP values with IoU threshold 0.1:0.1:0.9 are reported. † indicates the result is our re-implementation with the publicly available code.

| Method | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|--------------|------|------|------|------|------|------|------|------|------|
| STPN [21] | 52.0 | 44.7 | 35.5 | 25.8 | 16.9 | 9.9 | 4.3 | 1.2 | 0.1 |
| STPN+TP | 53.2 | 46.1 | 36.8 | 27.4 | 18.0 | 11.2 | 5.8 | 2.2 | 0.3 |
| Abs. Improve | +1.2 | +1.4 | +1.3 | +1.6 | +1.1 | +1.3 | +1.5 | +1.0 | +0.2 |
| A2CL-PT [19] | 61.2 | 56.1 | 48.1 | 39.0 | 30.1 | 19.2 | 10.6 | 4.8 | 1.0 |
| A2CL-PT+TP | 63.1 | 58.1 | 49.6 | 40.4 | 31.5 | 19.3 | 10.9 | 5.1 | 1.0 |
| Abs. Improve | +1.9 | +2.0 | +1.5 | +1.4 | +1.4 | +0.1 | +0.3 | +0.3 | +0.0 |
| BasNet [13] | 58.2 | 52.3 | 44.6 | 36.0 | 27.0 | 18.6 | 10.4 | 3.9 | 0.5 |
| BasNet+TP | 60.3 | 53.6 | 45.7 | 37.1 | 28.5 | 19.8 | 11.4 | 3.7 | 0.5 |
| Abs. Improve | +2.1 | +1.3 | +1.1 | +1.1 | +1.5 | +1.2 | 1.0 | -0.2 | +0.0 |
| WUM [14] | 67.5 | 61.2 | 52.3 | 43.4 | 33.7 | 22.9 | 12.1 | 3.9 | 0.5 |
| WUM+TP | 67.6 | 61.1 | 53.4 | 43.4 | 34.3 | 24.7 | 13.7 | 4.5 | 0.7 |
| Abs. Improve | +0.1 | -0.1 | +1.1 | +0.0 | +0.6 | +1.8 | +1.6 | +0.6 | +0.2 |
| WUM† [14] | 64.9 | 59.2 | 50.8 | 41.8 | 32.6 | 21.0 | 10.6 | 3.1 | 0.3 |
| WUM†+TP | 66.0 | 60.3 | 52.4 | 43.5 | 34.6 | 23.7 | 12.6 | 4.3 | 0.6 |
| Abs. Improve | +1.1 | +1.1 | +1.6 | +1.7 | +2.0 | +2.7 | +2.0 | +1.2 | +0.3 |

Table 2: The deconfounder applied to state-of-the-art WTAL models on ActivityNet-1.3 dataset. mAP values with IoU threshold 0.5, 0.75, and 0.95 are reported. The column AVG denotes the average mAP values at IoU thresholds 0.5:0.05:0.95. * indicates that the reported results are not present in the original paper.

| Method | 0.5 | 0.75 | 0.95 | AVG |
|--------------|------|------|------|-------|
| STPN [21] | 29.3 | 16.9 | 2.6 | 15.9* |
| STPN+TP | 31.3 | 18.0 | 3.6 | 17.7 |
| Abs. Improve | +2.0 | +1.1 | +1.0 | +1.8 |
| A2CL-PT [19] | 36.8 | 22.0 | 5.2 | 22.5 |
| A2CL-PT+TP | 37.4 | 23.5 | 5.9 | 23.7 |
| Abs. Improve | +0.6 | +1.5 | +0.7 | +1.2 |

PT². It can be observed that A2CL-PT+TP outperforms other state-of-the-art weakly-supervised models. Particularly, A2CL-PT+TP surpasses WUM by 0.4% on mAP with IoU=0.5. Moreover, A2CL-PT+TP achieves comparative performance with the fully supervised model CDC [34].

4.4. Ablation Studies

We further investigate the contributions of different components in the TS-PCA deconfounder in detail. We conduct ablation studies on THUMOS-14 dataset. Without loss of generality, we incorporate WUM with TS-PCA as the basic model to verify the effectiveness of different components. Specifically, the ablation studies aim to answer two questions. **Q1:** *What’s the contribution of each loss component?* **Q2:** *How many feature projectors are needed to encode \hat{z} ?*

²We choose A2CL-PT rather than WUM since WUM’s source code and pretrain-ed model on ActivityNet-1.3 are not publicly available.

4.4.1 Contributions of Different Loss Components

To get a better understanding of the proposed TS-PCA deconfounder, we further evaluate the key components of \mathcal{L}_{PCA} . **TS-PCA-R:** We discard the reconstruction loss, which is to encourage the reconstruction of video features based on the learned projectors, helpful for capturing the distribution of the input data. **TS-PCA-S:** The smoothing loss is to penalize the feature projection differences of consecutive segments. Here, we discard the smoothing loss to verify its contributions.

As shown in Table 5, WUM deployed with TS-PCA outperforms all its variants, namely TS-PCA-R and TS-PCA-S, and achieves an average improvement of 0.58% and 0.38%, respectively. Therefore, the reconstruction loss and the smoothing loss are both effective and necessary.

4.4.2 Performance over Different Number of Feature Projectors

Recall that TS-PCA learned L feature projectors to learn \hat{z} and reconstruct x . To investigate the performance of TS-PCA over different projector settings, we show the performance with respect to the number of projectors in Table 6. The results show that projecting each input video feature onto the first 5 principal components can preserve the data’s variation to a large extent. When we add more feature projectors, noise may be introduced. Therefore, the number of feature projectors is set to 5.

4.5. False Positive Analysis

Following DETAD [1], we utilize BasNet [13] on THUMOS-14 as the baseline model to conduct the false positive analysis [1] on *Over-completeness*, *Incompleteness*, and *Confusion*. We make detailed analysis on the top-

Table 3: Comparisons with the state-of-the-art methods on the testing set of THUMOS-14. We denote fully-supervised and weakly-supervised as Full and Weak, respectively. +TP represents the proposed deconfounder. mAP values with IoU threshold 0.1:0.1:0.9 are reported.

| Supervision | Method | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|-------------|--------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|------------|
| Full | R-C3D [39] | 54.5 | 51.5 | 44.8 | 35.6 | 28.9 | - | - | - | - |
| | SSN [46] | 66.0 | 59.4 | 51.9 | 41.0 | 29.8 | - | - | - | - |
| | MGG [17] | - | - | 53.9 | 46.8 | 37.4 | 29.5 | 21.3 | - | - |
| | GTAN [18] | 69.1 | 63.7 | 57.8 | 47.2 | 38.8 | - | - | - | - |
| | P-GCN [43] | 69.5 | 67.8 | 63.6 | 57.8 | 49.1 | - | - | - | - |
| | G-TAD [40] | - | - | 66.4 | 60.4 | 51.6 | 37.6 | 22.9 | - | - |
| Weak | UntrimmedNets [37] | 44.4 | 37.7 | 28.2 | 21.1 | 13.7 | - | - | - | - |
| | Hide-and-seek [36] | 36.4 | 27.8 | 19.5 | 12.7 | 6.8 | - | - | - | - |
| | AutoLoc [35] | - | - | 35.8 | 29.0 | 21.2 | 13.4 | 5.8 | - | - |
| | MAAN [41] | 59.8 | 50.8 | 41.1 | 30.6 | 20.3 | 12.0 | 6.9 | 2.6 | 0.2 |
| | W-TALC [24] | 55.2 | 49.6 | 40.1 | 31.1 | 22.8 | - | 7.6 | - | - |
| | DGAM [33] | 60.0 | 54.2 | 46.8 | 38.2 | 28.8 | 19.8 | 11.4 | 3.6 | 0.4 |
| | TSCN [44] | 63.4 | 57.6 | 47.8 | 37.7 | 28.7 | 19.4 | 10.2 | 3.9 | 0.7 |
| | A2CL-PT [19] | 61.2 | 56.1 | 48.1 | 39.0 | 30.1 | 19.2 | 10.6 | 4.8 | 1.0 |
| | WUM [14] | 67.5 | 61.2 | 52.3 | 43.4 | 33.7 | 22.9 | 12.1 | 3.9 | 0.5 |
| | WUM+TP (Ours) | 67.6 | 61.1 | 53.4 | 43.4 | 34.3 | 24.7 | 13.7 | 4.5 | 0.7 |

Table 4: Comparisons with the state-of-the-art methods on the validation set of ActivityNet-1.3. We denote fully-supervised and weakly-supervised as Full and Weak, respectively. +TP represents the proposed deconfounder. mAP values with IoU threshold of 0.5, 0.75, and 0.95 are reported. The column AVG denotes the average mAP values with IoU thresholds 0.5:0.05:0.95. All methods employ I3D as the feature extractor. * indicates that the reported results are not present in the original paper.

| Supervision | Method | 0.5 | 0.75 | 0.95 | AVG |
|-------------|---------------------------|-------------|-------------|------------|-------------|
| Full | CDC [34] | 45.3 | 26.0 | 0.2 | 23.8 |
| | R-C3D [39] | 26.8 | - | - | 12.7 |
| | BSN [15] | 52.5 | 33.5 | 8.9 | 34.3 |
| | G-TAD [40] | 50.4 | 34.6 | 9.0 | 34.1 |
| Weak | STPN [21] | 29.3 | 16.9 | 2.6 | 15.9* |
| | MAAN [41] | 33.7 | 21.9 | 5.5 | - |
| | Nguyen <i>et al.</i> [22] | 36.4 | 19.2 | 2.9 | - |
| | TSCN [44] | 35.3 | 21.4 | 5.3 | 21.7 |
| | WUM [14] | 37.0 | 23.9 | 5.7 | 23.7 |
| | A2CL-PT [19] | 36.8 | 22.0 | 5.2 | 22.5 |
| | A2CL-PT+TP(Ours) | 37.4 | 23.5 | 5.9 | 23.7 |

Table 5: Performance comparisons of TS-PCA and its variants (TS-PCA-R and TS-PCA-S) deployed on WUM on the testing set of THUMOS-14. mAP values with IoU threshold 0.3:0.1:0.7 are reported.

| Method | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 |
|---------------|-------------|-------------|-------------|-------------|-------------|
| TS-PCA-R | 51.8 | 42.7 | 33.5 | 22.9 | 12.3 |
| TS-PCA-S | 51.9 | 43.0 | 33.7 | 23.1 | 12.5 |
| TS-PCA | 52.3 | 43.4 | 34.0 | 23.6 | 12.8 |

5G predictions, where G is the number of ground truth instances. To observe the trend of each error type, we split the top-5G predictions into five equal splits and investigate

Table 6: With WUM [14] as the baseline model, performance comparisons over different number of feature projectors on THUMOS-14 dataset. mAP values at IoU=0.5 is reported.

| # of Projectors | 2 | 5 | 10 | 15 | 20 |
|-----------------|------|-------------|------|------|------|
| mAP | 32.2 | 34.6 | 32.2 | 32.5 | 32.7 |

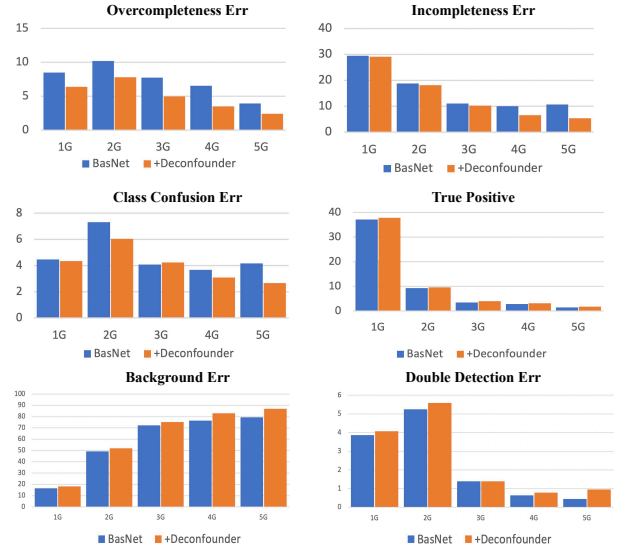


Figure 4: False positive analysis on the three error types, namely *Over-completeness*, *Incompleteness*, and *Confusion*. G is the number of ground truth instances.

the percentage of each error type. To highlight the comparisons, we reorganize the figures in Figure 4: BasNet+TP obtains lower error rates in all three types of errors when compared with BasNet. As we have minimized the three er-



Figure 5: Visualization of the predicted action instances on THUMOS-14. Three types of common issues: *Over-completeness* (Top), *Incompleteness* (Mid) and *Confusion* (Bottom), are illustrated. It could be observed that, with the deconfounder to calibrate the CAS scores generated by WTAL model, more accurate action instances are localized.

ror types caused by confounders, the rest of the error types that caused by the lack of full supervision become more significant compared to those in baseline. Note that the higher percentage of background error and double detection error do NOT mean that ours have more errors, because the total number of errors of ours is smaller than the baseline.

4.6. Qualitative Results

As mentioned in Section 1, *Over-completeness*, *Incompleteness*, and *Confusion* are the three types of localization errors frequently encountered in WTAL. The deconfounder theory infers a substitute confounder to calibrate conventional CAS, alleviating these problems. Figure 5 illustrates the qualitative results related to the above three common issues: 1) *Over-completeness* (Top): it’s hard for the WTAL model WUM to discriminate background from foreground. With the assistance of deconfounded strategy, false-positive background segments are largely reduced. 2) *Incompleteness* (Mid): the unobtrusive action segments are easily overlooked by WUM due to confounding bias. The deconfounded CAS could improve the completeness of action instances. 3) *Confusion* (Bottom): when two or more action categories such as CRICKETBOWLING and CRICK-

ETSHOT are visually correlated, it’s hard for the WTAL model to distinguish them. The deconfounded CAS can also tackle this challenge.

5. Conclusion

In this paper, we first summarized the three fundamental localization errors in WTAL. Then, we proposed a causal inference framework to identify that the reasons are due to the unobserved confounder. To capture the elusive confounder, we presented the unsupervised TS-PCA deconfounder, which exploits the unlabelled background to model an observed substitute for the confounder, to remove the confounding effect. Moreover, with a novel decoupled training scheme, the deconfounder is model-agnostic and could support any WTAL model in a plug-and-play fashion. Significant improvement on four state-of-the-art WTAL methods demonstrates the effectiveness and generalization ability of our deconfounded WTAL.

Acknowledgements. We thank the anonymous reviewers and ACs for their valuable comments. We thank Chong Chen at Damo Academy for his valuable suggestions on TS-PCA.

References

- [1] Humam Alwassel, Fabian Caba Heilbron, Victor Escorcia, and Bernard Ghanem. Diagnosing error in temporal action detectors. In *ECCV*, 2018. 6
- [2] Elias Bareinboim and Judea Pearl. Controlling selection bias in causal inference. In *Artificial Intelligence and Statistics*, 2012. 2
- [3] Michel Besserve, Arash Mehrjou, Rémy Sun, and Bernhard Schölkopf. Counterfactuals uncover the modular structure of deep generative models. *arXiv preprint arXiv:1812.03253*, 2018. 2
- [4] John M Burt Jr and Mark B Garman. Conditional monte carlo: A simulation technique for stochastic network analysis. *Management Science*, 18(3), 1971. 3
- [5] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 5
- [6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 5
- [7] Krzysztof Chalupka, Pietro Perona, and Frederick Eberhardt. Visual causal feature learning. *arXiv preprint arXiv:1412.2309*, 2014. 2
- [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. 1
- [9] Jingyuan Chen, Xuemeng Song, Liqiang Nie, Xiang Wang, Hanwang Zhang, and Tat-Seng Chua. Micro tells macro: Predicting the popularity of micro-videos via a transductive model. In *ACM MM*, 2016. 1
- [10] Linjiang Huang, Yan Huang, Wanli Ouyang, and Liang Wang. Relational prototypical network for weakly supervised temporal action localization. In *AAAI*, 2020. 2
- [11] Kosuke Imai and David A Van Dyk. Causal inference with general treatment regimes: Generalizing the propensity score. *Journal of the American Statistical Association*, 99(467), 2004. 4
- [12] Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes. In *ECCVW*, 2014. 5
- [13] Pilhyeon Lee, Youngjung Uh, and Hyeran Byun. Background suppression network for weakly-supervised temporal action localization. In *AAAI*, 2020. 1, 2, 3, 4, 5, 6
- [14] Pilhyeon Lee, Jinglu Wang, Yan Lu, and Hyeran Byun. Weakly-supervised temporal action localization by uncertainty modeling. *arXiv preprint arXiv:2006.07006*, 2020. 2, 5, 6, 7
- [15] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018. 1, 7
- [16] Daochang Liu, Tingting Jiang, and Yizhou Wang. Completeness modeling and context separation for weakly supervised temporal action localization. In *CVPR*, 2019. 1, 2
- [17] Yuan Liu, Lin Ma, Yifeng Zhang, Wei Liu, and Shih-Fu Chang. Multi-granularity generator for temporal action proposal. In *CVPR*, 2019. 1, 7
- [18] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Gaussian temporal awareness networks for action localization. In *CVPR*, 2019. 7
- [19] Kyle Min and Jason J Corso. Adversarial background-aware loss for weakly-supervised temporal activity localization. *arXiv preprint arXiv:2007.06643*, 2020. 5, 6, 7
- [20] Sanath Narayan, Hisham Cholakkal, Fahad Shahbaz Khan, and Ling Shao. 3c-net: Category count and center loss for weakly-supervised action localization. In *ICCV*, 2019. 2, 3, 4
- [21] Phuc Nguyen, Ting Liu, Gautam Prasad, and Bohyung Han. Weakly supervised action localization by sparse temporal pooling network. In *CVPR*, 2018. 1, 2, 5, 6, 7
- [22] Phuc Xuan Nguyen, Deva Ramanan, and Charles C Fowlkes. Weakly-supervised action localization with background modeling. In *ICCV*, 2019. 7
- [23] Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. Counterfactual vqa: A cause-effect look at language bias. *arXiv preprint arXiv:2006.04315*, 2020. 2
- [24] Sujoy Paul, Sourya Roy, Amit K Roy-Chowdhury, and Y. W-talc: Weakly-supervised temporal activity localization and classification. In *ECCV*, 2018. 2, 3, 4, 5, 7
- [25] Judea Pearl. Causality: Models, reasoning, and inference. *Springer*, 2000. 2
- [26] Judea Pearl. *Causality*. Cambridge university press, 2009. 2, 3
- [27] Judea Pearl. Interpretation and identification of causal mediation. *Psychological methods*, 19(4), 2014. 2
- [28] Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016. 2
- [29] Jiaxin Qi, Yulei Niu, Jianqiang Huang, and Hanwang Zhang. Two causal principles for improving visual dialog. In *CVPR*, 2020. 2
- [30] Paul R Rosenbaum and Donald B Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1), 1983. 4
- [31] Donald B Rubin. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469), 2005. 2, 3
- [32] Donald B Rubin. Essential concepts of causal inference: a remarkable history and an intriguing future. *Biostatistics & Epidemiology*, 3(1), 2019. 2
- [33] Baifeng Shi, Qi Dai, Yadong Mu, and Jingdong Wang. Weakly-supervised action localization by generative attention modeling. In *CVPR*, 2020. 2, 7
- [34] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. Cdc: Convolutional-deconvolutional networks for precise temporal action localization in untrimmed videos. In *CVPR*, 2017. 6, 7
- [35] Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. Autoloc: Weaklysupervised temporal action localization in untrimmed videos. In *ECCV*, 2018. 1, 2, 7

- [36] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*, 2017. 2, 7
- [37] Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. Untrimmednets for weakly supervised action recognition and detection. In *CVPR*, 2017. 1, 2, 7
- [38] Yixin Wang and David M Blei. The blessings of multiple causes. *Journal of the American Statistical Association*, 114(528), 2019. 2, 3
- [39] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: region convolutional 3d network for temporal activity detection. In *ICCV*, 2017. 7
- [40] Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 7
- [41] Yuan Yuan, Yueming Lyu, Xi Shen, Ivor W Tsang, and Dit-Yan Yeung. Marginalized average attentional network for weakly-supervised learning. *arXiv preprint arXiv:1905.08586*, 2019. 2, 5, 7
- [42] Zhongqi Yue, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Interventional few-shot learning. *NeurIPS*, 33, 2020. 2
- [43] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph convolutional networks for temporal action localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. 7
- [44] Yuanhao Zhai, Le Wang, Wei Tang, Qilin Zhang, Junsong Yuan, and Gang Hua. Two-stream consensus network for weakly-supervised temporal action localization. In *ECCV*. Springer, 2020. 7
- [45] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xiansheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. *arXiv preprint arXiv:2009.12547*, 2020. 2
- [46] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *ICCV*, 2017. 5, 7
- [47] Jia-Xing Zhong, Nannan Li, Weijie Kong, Tao Zhang, Thomas H Li, and Ge Li. Step-by-step erasing, one-by-one collection: a weakly supervised temporal action detector. In *ACM MM*, 2018. 2