Set-Supervised Action Learning in Procedural Task Videos via Pairwise Order Consistency

Zijia Lu
Northeastern Univeristy
lu.zij@northeastern.edu

Ehsan Elhamifar
Northeastern University
e.elhamifar@northeastern.edu

Abstract

We address the problem of set-supervised action learning, whose goal is to learn an action segmentation model using weak supervision in the form of sets of actions occurring in training videos. Our key observation is that videos within the same task have similar ordering of actions, which can be leveraged for effective learning. Therefore, we propose an attention-based method with a new Pairwise Ordering Consistency (POC) loss that encourages that for each common action pair in two videos of the same task, the attentions of actions follow a similar ordering. Unlike existing sequence alignment methods, which misalign actions in videos with different orderings or cannot reliably separate more from less consistent orderings, our POC loss efficiently aligns videos with different action orders and is differentiable, which enables end-to-end training. In addition, it avoids the time-consuming pseudo-label generation of prior works. Our method efficiently learns the actions and their temporal locations, therefore, extends the existing attention-based action localization methods from learning one action per video to multiple actions using our POC loss along with video-level and frame-level losses. By experiments on three datasets, we demonstrate that our method significantly improves the state of the art. We also show that our method, with a small modification, can effectively address the transcript-supervised action learning task, where actions and their ordering are available during training.[1]

1. Introduction

Learning actions by partitioning long and untrimmed procedural videos into action segments has recently drawn increasing attention in video understanding. Fully-supervised methods [23, 26, 46, 55, 57, 62] require dense annotation of training videos with framewise action labels, which is costly and unscalable. Therefore, weakly-supervised methods [5, 9, 13, 30, 31, 33, 37, 44, 45, 52, 58] learn from weak labels, in the form of *action transcripts*, i.e., ordered lists of actions in videos, or *action sets*, i.e., sets of unique actions in videos (obtained from video narrations, captions or meta-tags). For action transcripts/sets, learning faces the major challenge of finding the temporal regions of actions in videos during the training. Learning from action sets, referred to as set-supervised action learning, additionally faces the challenge of not knowing the ground-truth action orderings in training videos.

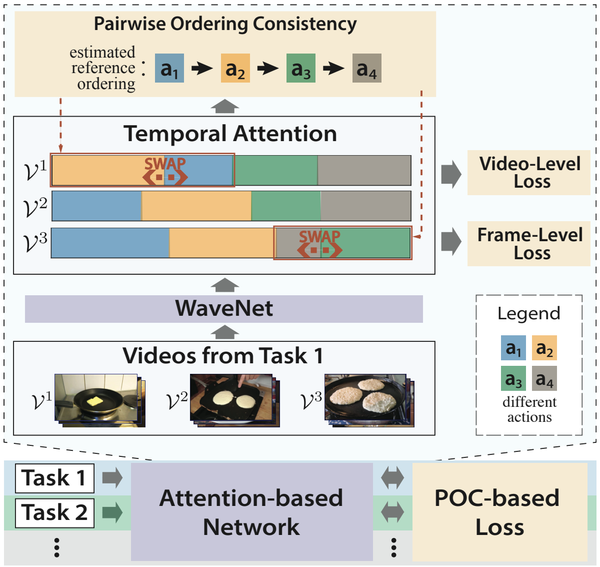


Figure 1. Our attention-based framework with pairwise ordering consistency for set-supervised action learning.

Most prior works on set-supervised action learning [31, 33, 44] alternate between three steps: i) generating action-transcripts from action-sets, ii) generating pseudo-labels from transcripts using Viterbi decoding, iii) training the model using pseudo-labels. However, generating transcripts and pseudo-labels is costly, slowing down the training and inference speeds, while the often erroneous pseudo-labels in early training iterations degrade the performance. [13] proposes a two-branch CNN that co-supervise each other and directly predicts the label and length of each video segment. Despite faster training/inference speed, [13] uses a fixed ra-

[1] Code available at https://github.com/ZijiaLewisLu/CVPR22-POC.

tio of the number of segments to the video length, which cannot handle dense action regions, i.e., it may undersegment short videos that contain many actions. More importantly, it predicts the labels of segments independently, which ignores the ordering between actions and similar transcripts of training videos of each task.

In fact, videos of the same task often have similar transcripts. For example, in videos of 'make fried-egg', the step of 'cracking egg' is followed by 'frying egg'. While such similar ordering has been the key in the development and success of unsupervised action learning methods [1, 11, 12, 15, 25, 53, 64], it has not been exploited for set-supervised action learning. On the other hand, temporal attention is a powerful mechanism for finding temporal regions of actions in videos. However, existing works based on attention [27, 28, 38–40, 54, 63], assume one action is present in the video, which must be distinguished from the background frames. Therefore, extending temporal attention to videos of complex tasks with multiple actions and regulating it according to ordering consistency within and across videos remains a major challenge.

Paper Contributions. In this paper, we propose a temporal attention-based method for set-supervised action learning by leveraging the similarity of action ordering of videos of each task, see Figure 1. We design a new loss function, referred to as Pairwise Ordering Consistency (POC) loss, which encourages that for each common pair of actions in videos of the same task, attention predictions follow the same ordering across videos. Our work has the following advantages with respect to the state of the art:

– Our new POC loss resolves drawbacks of Dynamic Time Warping (DTW) [48] and Edit Distance (ED) [29]. Unlike DTW, which misaligns actions when applied to videos with different transcripts (action orderings), our method correctly aligns actions and brings the representations of the same action closer to each other. Unlike ED, which may not properly distinguish different inconsistent orderings from one another, our method efficiently distinguishes different consistency levels of action orderings by comparing the percentage of inconsistent action pairs. Additionally, our POC loss is differentiable, enabling feature learning, and its computational complexity is linear in the number of videos, thereby is scalable to large training sets.

– Unlike three-step approaches [31, 33, 44], our method has a single step of minimizing a new loss, hence, enjoys faster training/inference and does not suffer from error amplification. Unlike [13], our method uses similarity of a task's videos, gives frame predictions, handles short videos with many actions and has much smaller number of parameters.

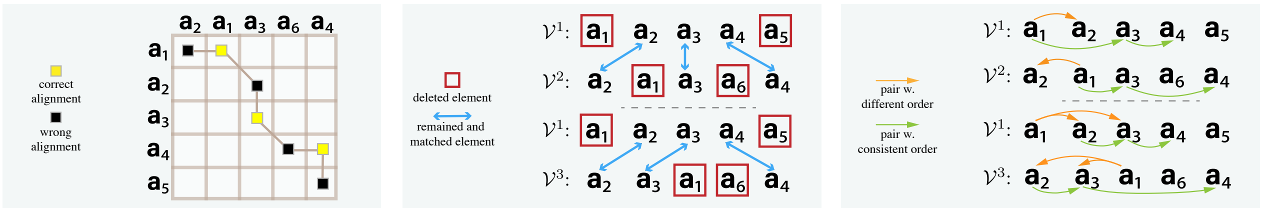
– We extend the existing works on attention-based action localization, from one to multiple actions in each videos. Our POC loss penalizes the overlap between attentions of different actions for better localization and enforces consistent ordering between actions, which has not been addressed in prior works on attention-based action localization. Unlike existing works, our method does not assume that the length of each action is a fixed ratio of the total number of video frames and alternatively learns it via video-level action recognition, where the attention of an action is enforced to cover every possible frame of it.

– Last, but not least, by extensive experiments on three datasets, we show that our proposed method outperforms existing set-supervised algorithms. We show that, with a small modification, our method can also effectively address transcript-supervised action learning, where the actions and their orderings are available during training.

2. Related Works

Action Learning with Set Supervision. Many long untrimmed videos [7, 14, 41, 51, 56, 64] and the high cost of gathering framewise action annotations have motivated many works on localization and classification of actions using minimum supervision. Set-supervised methods learn a model from the ground-truth sets of actions in training videos. [44] generates video transcripts from action-sets and runs Viterbi decoding based on HMM to produce framewise action labels. [31] proposed a Set-Constrained Viterbi algorithm to ensure the optimal segmentation covers all actions in the action-set. Building on this idea, [33] treats the most confident frames of each action as anchors to exclude unreliable segmentations to improve the training complexity and performance. On the other hand, [13] directly predicts actions and their lengths without Viterbi decoding to improve the inference speed. Since action-sets have no information about localization, consistent action ordering is an important assumption to improve action learning. [31,33,44] learn ordering via the transition model of an HMM. However, Viterbi decoding results in large computational cost. Our method does not use Viterbi decoding and learns consistent orderings by enforcing pairwise ordering similarity.

Action Learning with Other Weak Supervision. Some works have studied action segmentation using video transcripts [2, 5, 9, 18, 30, 31, 37, 44, 45, 64] or video summaries [42, 61]. Earlier works exploit using transcripts via speech recognition techniques [24, 43], two-step optimization schemes [20, 21, 24, 43], connectionist temporal classification [18, 36] and iteratively boundary refinement [9]. Also, [5, 30, 45] perform Viterbi decoding between a video and its transcript. [58] studies a two-branch CNN with co-supervision between branches. [37] models actions using low-dimensional subspaces, which allows adjusting subspace dimensions based on complexities of actions. To remove or further reduce the need for annotation, [1, 11, 12, 16, 25, 32, 49, 50, 53] have studied unsupervised segmentation by leveraging the shared structure of


correct alignment
wrong alignment
a_2 a_1 a_3 a_6 a_4
a_1
a_2
a_3
a_4
a_5
deleted element
remained and matched element
$\mathcal{V}^1$: a_1 a_2 a_3 a_4 a_5
$\mathcal{V}^2$: a_2 a_1 a_3 a_6 a_4
$\mathcal{V}^1$: a_1 a_2 a_3 a_4 a_5
$\mathcal{V}^3$: a_2 a_3 a_1 a_6 a_4
pair w. different order
pair w. consistent order
$\mathcal{V}^1$: a_1 a_2 a_3 a_4 a_5
$\mathcal{V}^2$: a_2 a_1 a_3 a_6 a_4
$\mathcal{V}^1$: a_1 a_2 a_3 a_4 a_5
$\mathcal{V}^3$: a_2 a_3 a_1 a_6 a_4

Dynamic Time Warping **Edit Distance** **Pairwise Ordering Consistency**

Figure 2. Illustration of Dynamic Time Warping, Edit Distance and Pairwise Ordering Consistency applied on videos with different action orderings.

videos from similar tasks, while the recent work in [52] has studied the new setup of semi-weakly-supervised learning. Also, [35] has explored timestamp supervision, where each action is annotated with one frame.

Sequence Alignment Distances. Two major classes of sequence alignment methods are Dynamic Time Warping [48] and Edit Distance [29]. DTW finds element-wise ordered alignment between two sequences and has been generalized to a differentiable setting [5,6,17]. However, standard DTW aligns sequences by assuming that they follow the same ordering. Therefore, it leads to misalignment between actions when applied to videos with different action orderings, as shown in Figure 2 (left). [10, 53] allow some elements to stay unmatched, yet the unmatched elements and their relative orderings are ignored in optimizing the alignment and cannot be improved.

ED measures the pairwise sequence distance as the minimum number of edit operations to transform one sequence into the other and has been generalized to a differentiable distance in [19]. However, ED may not distinguish more from less consistent orderings under certain scenarios. For example, in Figure 2 (middle), the ordering between $\{a_1, a_2, a_3\}$ is more consistent between videos $(\mathcal{V}_1, \mathcal{V}_2)$ than $(\mathcal{V}_1, \mathcal{V}_3)$ as the first only has one violating action pair, (a_1, a_2), and the second has two violating pairs, (a_1, a_2) and (a_1, a_3). However, ED of both video pairs is 4 (the number of deleted elements), thus fails to capture the differences in order consistencies of these video pairs. Also, ED incurs a cost if two videos contain uncommon actions, such as $\{a_5, a_6\}$ in our toy example. Minimizing ED implies videos must have the same action-sets, while in fact videos of the same task can have slightly different action-sets. Finally, [8] extends ED to allow for transposition between adjacent elements, yet it is not differentiable. In contrast, our POC loss is differentiable, respects the difference in action-sets and can accurately measure the ordering discrepancy by comparing the ordering between action pairs.

3. Set-supervised Action Learning via Pairwise Order Consistency

3.1. Problem Statement

Assume we have N videos of various tasks, e.g., from making different recipes in the cooking domain. For each video $\mathcal{V}^i$ of length T^i, we have a set of pre-extracted frame-wise features $\mathcal{X}^i = \{\boldsymbol{x}_1^i, \ldots, \boldsymbol{x}_{T^i}^i\}$ and the action-set $\mathcal{A}^i = \{a_1^i, ..., a_{|\mathcal{A}^i|}^i\}$. The action-set contains the list of actions occurring in the video with $a_j^i \in \{1, 2, \ldots, A\}$. Here, $|\mathcal{A}^i|$ denotes the number of actions in the i-th video and A denotes the total number of actions across all videos. Present actions in the action-set, $a \in \mathcal{A}^i$, are often referred to as positive actions and the others as negative actions. For simplicity of notation, we drop the superscript i (when referring to video i) when it is clear from the context. The goal of set-supervised action learning is to learn a segmentation model that predicts the action of each frame during testing, using training videos that are weakly-labeled by their action-sets.

3.2. Proposed Method

We propose an attention-based network with pairwise ordering consistency for set-supervised action learning. Our framework consists of the following components.

3.2.1 Feature Learning and Temporal Attention

We use a fully convolutional network for feature learning followed by A attention modules for localizing actions in videos. More specifically, we use a WaveNet backbone [60] for feature learning to capture the temporal information among frame features,

$$(\boldsymbol{h}_1, \ldots, \boldsymbol{h}_{T'}) = \text{WaveNet}\big((\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T)\big). \tag{1}$$

We have $T' \leq T$ when we use a MaxPooling layer. We use an attention module, $\mathcal{F}^{\text{att}}$, to compute the score/probability of each timestamp belonging to each action,

$$\begin{aligned} \boldsymbol{\Gamma} &= \mathcal{F}^{\text{att}}\big((\boldsymbol{h}_1, \ldots, \boldsymbol{h}_{T'})\big) \in \mathbb{R}^{A \times T'}, \\ \boldsymbol{W} &= \text{softmax}(\boldsymbol{\Gamma}) \in [0, 1]^{A \times T'}, \end{aligned} \tag{2}$$

where $\boldsymbol{\Gamma}$ and $\boldsymbol{W}$ denote, respectively, the unnormalized and normalized attention weights, where $\sum_a \boldsymbol{W}_{a,t} = 1$. The value of $\boldsymbol{W}_{a,t}$ being close to 1 indicates that frame t is likely to belong to action a. We compute the estimated length of action a as $\tau_a = \sum_t \boldsymbol{W}_{a,t} \in [0, T']$. To learn the parameters of our model, we propose a novel loss function that consists of several components as follows.

3.2.2 Pairwise Order Consistency (POC)

Motivation. A key observation in our work is that for videos of the same task, the pairwise ordering of common

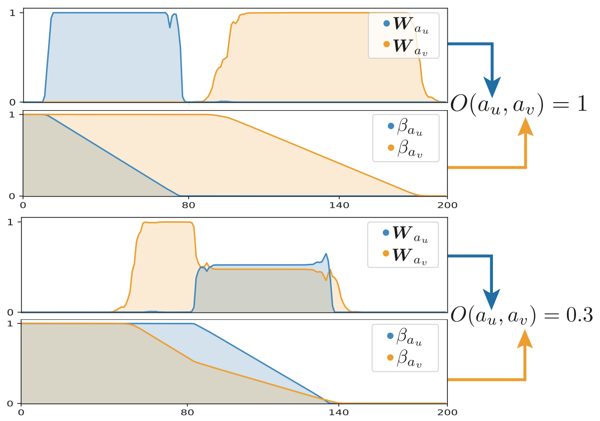


Figure 3. Illustration of our order score encoding the relative positions between actions. Top: $O(a_u, a_v) = 1$ as a_u occurs before a_v. Bottom: $O(a_u, a_v) = 0.3$ as a_u occurs after the start of a_v, yet it overlaps with the later part of a_v.

actions are similar. Indeed, for the videos of the same task in Breakfast [22], CrossTask [64] and MPII Cooking 2 [47] datasets, respectively, 88%, 66% and 67% of the common action pairs have consistent ordering. Therefore, our goal is to learn attention models of actions so that predictions for pairs of common actions in videos of each task follow a consistent ordering. To achieve the goal, we measure the ordering difference between two videos as the percentage of common action pairs that have an inconsistent order. For example, in Figure 2 (right), both $(\mathcal{V}^1, \mathcal{V}^2)$ and $(\mathcal{V}^1, \mathcal{V}^3)$ have 4 common actions $\{a_1, a_2, a_3, a_4\}$, thus 6 common action pairs. Notice that the ordering of $\mathcal{V}_2$ is more similar to $\mathcal{V}_1$ than $\mathcal{V}_3$ as $(\mathcal{V}^1, \mathcal{V}^2)$ only has one violating action pair (the ordering difference is $\frac{1}{6}$), while $(\mathcal{V}^1, \mathcal{V}^3)$ has two violating pairs (the ordering difference is $\frac{1}{3}$). To allow for different action-sets, we do not consider the ordering of the uncommon actions, such as $\{a_5, a_6\}$. Therefore, unlike DTW, we can effectively handle videos with different orderings. Unlike ED, we efficiently separate the more from less consistent orderings by comparing the pairwise action orders while respecting the difference between action-sets.

A main challenge is how to measure if an action a_u occurs before an action a_v using attention outputs, which are probabilities. To tackle this, we propose to find the ordering score of (a_u, a_v), denoted by $O(a_u, a_v)$, as the probability of a_u occurring before a_v. To do so, first we compute

$$\beta_{a_v,t} = \frac{1}{\tau_{a_v}} \sum_{k=t+1}^{T'} \boldsymbol{W}_{a_v,k} \in [0, 1], \tag{3}$$

which measures the probability of action a_v starting after time t. We then compute the ordering score as

$$O(a_u, a_v) = \frac{1}{\tau_{a_u}} \sum_{t} \boldsymbol{W}_{a_u,t} \beta_{a_v,t} \in [0, 1]. \tag{4}$$

For example, given two attention outputs for actions a_u and a_v shown in Figure 3 (top), $\beta_{a_v,t}$ would be 1 for the first 80 frames, during which action a_u started and finished. Therefore, $O(a_u, a_v)$ according to (4) would be 1, as desired. On the other hand, in Figure 3 (bottom), a_u occurs from frame 80 to 140, after the start of a_v, yet overlaps with the later part of a_v. Thus, $O(a_v, a_u) = 0.3$ is a small but non-zero value, indicating their relative ordering and overlap.

POC Loss. Given the output of the attentions for video i, we compute an ordering score between each pair of its actions. More specifically, for $a_u, a_v \in \mathcal{A}^i$, we compute the ordering score $O^i(a_u, a_v) \in [0, 1]$, which is the probability of a_u happening before a_v. Our goal is to ensure that the ordering of a_u, a_v would be consistent across all videos that contain this pair of actions. Let $\Lambda(a_u, a_v)$ denote the set of all videos that have both a_u and a_v in their action sets. We define a reference ordering score between (a_u, a_v) by computing the average of $O^i(a_u, a_v)$ across all videos,

$$O^\star(a_u, a_v) = \frac{1}{|\Lambda(a_u, a_v)|} \sum_{i \in \Lambda(a_u, a_v)} O^i(a_u, a_v). \tag{5}$$

We define the discrepancy between the ordering in each video i and the reference as

$$\begin{aligned} \pi^i(a_u, a_v) = 1 - O^i(a_u, a_v) O^\star(a_u, a_v) \\ - O^i(a_v, a_u) O^\star(a_v, a_u). \end{aligned} \tag{6}$$

Notice that π^i will be close to 0 when the attention prediction of a_u is before/after a_v in both the video and the reference ordering, while π^i would be close to one when the ordering in the video and the reference disagree. Finally, we define the Pairwise Order Consistency (POC) loss as

$$\mathcal{L}_{\text{poc}}(\{\mathcal{V}^i\}) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|\mathcal{A}^i|^2} \sum_{a_u, a_v \in \mathcal{A}^i} \pi^i(a_u, a_v), \tag{7}$$

which measures the aggregated discrepancy between action ordering between each video and the reference. As a result, minimizing this loss not only enforces a consistent ordering of common action pairs across videos, but also reduces the overlap between action attentions. Thanks to the differentiability of our loss, we also obtain an end-to-end framework for learning actions, therefore avoiding the high computational complexity of the Viterbi decoding and pseudo-label generation as in the prior works [31, 33, 44]. Our POC loss is directly computed from attentions and minimizing it guides the learning of attention modules and the network parameters. Notice also that POC loss can be applied to any networks that predict framewise action labels of videos.

Remark 1 *A main advantage of computing a reference ordering as in (5) and computing the loss as in (7) is linear computational complexity with respect to the number of* *videos. Alternatively, we could compute the discrepancy between every pair of videos. However, this leads to quadratic complexity with respect to the number of videos (similar to DTW and ED), and performed worse in our experiments than using the reference ordering.*

In the supplementary materials, we discuss that our loss can handle repeated actions and varied action ordering and provides more analysis of computation complexity and the effect of using the reference ordering on the optimal solution.

3.2.3 Frame-Level and Video-Level Losses

For action recognition, we will use the attention features to build video-level action features that will be given to a classifier, which will be learned using video-level losses. Additionally, we regularize the learning of the classifier by defining frame-level losses on the attentions.

Frame-Level Losses. We use a framewise attention ranking loss to ensure that, for each training video, the attentions of positive actions are higher than those of negative actions at each time instant,

$$\mathcal{L}_{\text{f-rk}}(\mathcal{V}) = \frac{1}{T'}\sum_t \log\Big(1+\sum_{a\in\mathcal{A},a'\notin\mathcal{A}} e^{(\boldsymbol{\Gamma}_{a',t}-\boldsymbol{\Gamma}_{a,t})}\Big). \quad (8)$$

In procedural videos, some frames do not belong to any action, hence referred to as background[2] frames. Therefore, we also use a 'background' action class. Previous works [10, 25, 53] have shown that regularizing the percentage of background frames has a large impact on the performance. Thus, we propose a background length loss,

$$\mathcal{L}_{\text{f-bg}}(\mathcal{V}) = \begin{cases} \log(\tau_{\min}/\tau_{\text{bg}}), & \text{if } \tau_{\text{bg}} < \tau_{\min}, \\ 0, & \text{if } \tau_{\min} \leq \tau_{\text{bg}} \leq \tau_{\max}, \\ \log(\tau_{\text{bg}}/\tau_{\max}), & \text{if } \tau_{\text{bg}} > \tau_{\max}, \end{cases} \quad (9)$$

which promotes that $\tau_{\text{bg}} \triangleq \sum_t \boldsymbol{W}_{t,a_{\text{bg}}}$, which is the estimated length of the background action using the associated attention module, will be between the two predefined thresholds of $\tau_{\min}$ and $\tau_{\max}$.

A challenge when learning temporal attention is that the predicted frames for an action could be scattered in the video, instead of forming one or a few segments. To overcome this, we propose a centering loss

$$\mathcal{L}_{\text{f-cr}}(\mathcal{V}) = \frac{1}{|\mathcal{A}|\cdot T'}\sum_{a\in\mathcal{A},a\neq a_{\text{bg}}}\sum_t \boldsymbol{W}_{a,t}(t-\bar{t}_a)^2, \quad (10)$$

where $\bar{t}_a \triangleq (\sum_t t\cdot \boldsymbol{W}_{a,t})/(\sum_t \boldsymbol{W}_{a,t})$ is the temporal center of action a.[3] This loss imposes a large penalty if $\boldsymbol{W}_{a,t}$ is large for frames that are far from the action temporal center.

Video-Level Losses. For action recognition, we build a video-level feature for each action a using its attention,

$$\boldsymbol{g}_a = \frac{1}{T'}\sum_t \boldsymbol{W}_{a,t}\boldsymbol{h}_t, \quad (11)$$

which will be fed into an action classifier $\mathcal{F}^{\text{cls}}$. We consider two losses for supervising the action classifier [34]. First, we use a ranking loss,

$$\mathcal{L}_{\text{v-rk}}(\mathcal{V}) = \log\Big(1+\sum_{a\in\mathcal{A},a'\notin\mathcal{A}} e^{(\mathcal{F}^{\text{cls}}(\boldsymbol{g}_{a'})-\mathcal{F}^{\text{cls}}(\boldsymbol{g}_a))}\Big), \quad (12)$$

where $\mathcal{F}^{\text{cls}}(\boldsymbol{g}_a)$ is the probability logit without passing it through the sigmoid function. Since set-supervised action recognition is a multi-label learning problem, the ranking loss resolves the positive and negative sample imbalance. However, its drawback is requiring to tune a threshold on the probability logit to separate the positive from negative actions. To resolve this issue, we use a binary cross-entropy loss with a dynamically estimated action threshold,

$$\begin{aligned} p_a &= \text{sigmoid}\left(\mathcal{F}^{\text{cls}}(\boldsymbol{g}_a) - \mathcal{F}^{\text{thr}}(\boldsymbol{g}_a)\right), \\ \mathcal{L}_{\text{v-ce}}(\mathcal{V}) &= -\sum_{a\in\mathcal{A}}\log(p_a) - \sum_{a'\notin\mathcal{A}}\log(1-p_{a'}). \end{aligned} \quad (13)$$

Here, $\mathcal{F}^{\text{thr}}(\boldsymbol{g}_a)$ is the estimated threshold for action a via a sub-network $\mathcal{F}^{\text{thr}}$, which is then used to compute the action probability p_a.

3.2.4 Training and Inference

Our final loss function is the weighted sum of the POC, frame-level and video-level losses, discussed above. In each iteration of the training, we sample a batch of videos from the same task to learn action ordering. We do not consider the ordering between 'background' and other actions as background can occur anywhere. For testing, we consider two scenarios: action segmentation, where we assume the action-sets of test videos are unknown, and action alignment, where the action-sets are known. For action segmentation, we first estimate the action-set of a test videos as $\hat{\mathcal{A}} = \{a|p_a \geq 0.5\}$. We obtain the action label $\hat{y}_t$ of frame t by finding the maximum attention value over $\hat{\mathcal{A}}$, i.e., $\hat{y}_t = \text{argmax}_{a\in\hat{\mathcal{A}}}\boldsymbol{W}_{a,t}$. For action alignment, we directly compute the argmax over the ground-truth $\mathcal{A}$, which is assumed to be given.

4. Experiments

We evaluate the performance of our proposed method for action segmentation and action alignment tasks, against prior set-supervised methods. We also compare with the state-of-the-art attention-based action localization method

[2] On average, 74.8%, 29.7% and 7.3% of frames of a video in CrossTask, MPII Cooking2 and Breakfast, respectively, are background.

[3] $\mathcal{L}_{\text{f-cr}}$ can also handle multiple occurrences of an action by using multiple temporal centers (e.g., via KMeans). However, we found that using one center works well in our experiments.

UM [28]. Additionally, we demonstrate that a simple modification of our method can address transcript-supervised action learning, where we know the transcripts (sequence of actions) in videos during the training time. We also perform ablation studies to investigate the effect of each of our proposed loss functions, compare the performance of POC loss under different scenarios and show qualitative results for video segmentation using our method.

4.1. Experimental Setup

Datasets. We use three video datasets for evaluations. The *Breakfast* [22] dataset contains 1,712 videos of 10 cooking activities with 48 different actions. On average, each video has 5.2 unique actions and 6.9 action segments, as 10% of the actions have multiple occurrences. The *CrossTask* [64] dataset consists of videos from 18 tasks. Following [37], we train on the 14 cooking-related tasks, which have 2,522 videos and 80 actions. On average each video has 6.1 actions and 14.4 action segments, with 25% actions having multiple occurrences. The *MPII Cooking 2* (MC2) [47] is a smaller dataset with 273 videos from 74 recipes and 68 actions. On average each video has 16.7 actions and 95.2 action segments, with 50% of actions having multiple occurrences. The datasets become more challenging from Breakfast to CrossTask and then to MC2, as their videos have more segments, action repetitions and background frames.

Evaluation Metrics. We use three evaluation metrics: *1) Mean-over-Frame* (MoF), which is the percentage of frames whose action labels are predicted correctly. *2) Intersection over Union* (IoU), defined as $\frac{1}{A}\sum_a |GT_a \cap D_a|/|GT_a \cup D_a|$ where GT_a and D_a are the ground-truth and predicted set of frames for action a. We follow the implementation in [37]. *3) Midpoint Hit Metric* (MidH), as the percentage of predicted action segments whose middle frames belong to the ground-truth segment of the action.

Implementation Details. Consistent with prior works, we use the released four train/test splits for Breakfast and the one train/test split for CrossTask and MC2. Meanwhile, we use the I3D features [4] extracted by [13] on Breakfast and the released 3,200-dimension features on CrossTask. Since [13] has not released their extracted I3D features from MC2, we extract I3D features ourselves.

To adapt our method to the transcript-supervised learning task, we will estimate the reference order $O^\star$ in (6) from the ground-truth transcripts instead of attentions of the videos. Following [30, 37, 45], for action alignment, we run Viterbi decoding between videos and the given transcripts to predict the framewise action label. Due to the nature of the weak supervision, the performance of all transcript/set-supervised models changes for different initializations. Most prior works only report the results of their *best run*, see [37, 59]. Similar to [37], we run our method for three runs and report the best run results for a fair comparison and the averaged results over runs for thorough evaluation. We include more performance statistics in the supplementary material.

4.2. Experimental Results

In Table 1, we report the *best run* results for the prior set-supervised action segmentation methods and the results of our POC for the best run and the average over three runs, denoted as *POC (best)* and *POC (average)*, respectively. Since the action localization method UM [28] does not evaluate on the datasets we use, we show the results of our replication using its released code, denoted by <u>UM</u>. Meanwhile, prior set-supervised action segmentation methods have not reported IoU and have not tested on CrossTask. For a fair comparison, we replicate the state-of-the-art model SCT [13] on all three datasets with its released code, whose result is denoted by <u>SCT</u>. Notice we also collect the scores of SCT on Breakfast and MC2 from its paper, but our replication obtain higher accuracy than those reported in that paper.

In Table 2, we show the *best run* results for prior transcript-supervised action segmentation methods and the best and average results of our method. MuCon [58] does not report the best run results; we include their reported average results and standard deviation. For the transcript-supervised segmentation, we do not compare on MC2 as no prior transcript-supervised method evaluated on the dataset.

Set-Supervised Action Learning. As Table 1 shows, POC outperforms the all set-supervised methods on the three datasets for both action segmentation and action alignment tasks. UM has low performance as it assumes each video contains only one action, thus its attention does not learn distinct locations for different actions. In contrast, our POC loss penalizes the overlap between attentions. Our method also exceeds SCT as it does not consider the ordering between action as we do in our POC loss.

Notice that on Breakfast, POC improves the MoF and IoU for action segmentation by *7.6%* and *16.2%*, respectively, over the second best results (SCT), showing the effectiveness of consistent action ordering in learning action localization. On CrossTask, our method improves over SCT by *6.7%* on MoF and *4.8%* on IoU on action segmentation. On MC2, although 50% of the actions in a video have multiple occurrences, our POC loss is still able to model the action orderings efficiently. We improve the MidH by *1.6%* on action segmentation and *3.5%* on action alignment. Without the POC loss, the MidH of our model drops by *5.3%* and *4.7%* for action segmentation and alignment, respectively.

On the other hand, the set-supervised pseudo-label based methods [31, 33, 44] have a big performance gap between action segmentation and action alignment tasks, e.g., ACV has a *11.7%* gap for MoF on Breakfast. In comparison, the gap of our POC at MoF is 3.9% on Breakfast and only *0.1%*

	Segmentation		Alignment	
Breakfast	MoF	IoU	MoF	IoU
Action Set [44]	23.3		28.4	
UM [28]	29.1	15.8	29.5	16.8
SCV [31]	30.2		40.8	
SCT [13]	30.4			
SCT [13]	34.8	17.3	37.9	19.2
ACV [33]	33.4		45.1	
POC (best)	**42.4**	**33.5**	**46.3**	**36.7**
POC (average)	40.1	32.5	43.6	35.8
CrossTask	MoF	IoU	MoF	IoU
UM [28]	35.8	10.5	40.3	13.0
SCT [13]	37.3	11.4	40.4	12.6
POC (best)	**44.0**	**16.2**	**44.1**	**16.2**
POC (average)	42.9	15.6	42.6	15.6
MC2	MidH	IoU	MidH	IoU
Action Set [44]	10.6		10.6	
UM [28]	12.1	3.9	12.5	5.1
SCV [31]	14.5		15.1	
SCT [13]	14.3			
SCT [13]	14.7	4.7	15.9	4.8
ACV [33]	15.5		16.2	
POC (best)	**17.1**	**9.0**	**19.7**	**10.0**
POC (average)	13.1	5.7	16.0	6.0

Table 1. Set-Supervised action segmentation and alignment results.

on CrossTask. This comes from the fact that pseudo-label based methods cannot directly predict the action-set of a test video, but compare a video with each training action-set to find the most likely one, which is often erroneous or only partially correct. Our method accurately predicts the action-sets of test videos, thanks to the action classifier learned with our video-level losses, hence boosts the performance of action segmentation. Thus, our set-supervised POC, despite not having access to ground-truth action orderings of training videos, achieves competitive results with transcript-supervised methods (compare results of POC in Table 1 with results of existing works in Table 2). POC with set supervision achieves a similar MoF as NNV and a similar IoU as CDFL on Breakfast and even surpasses the state-of-the-art TASL on CrossTask for action segmentation.

Transcript-Supervised Action Learning. In Table 2, we show POC trained with ground-truth transcripts achieves competitive results w.r.t. the state-of-the-art transcript-supervised methods, even obtaining the best results for some cases. For action segmentation, POC successfully exceeds TASL at IoU by 2.8% on Breakfast and 1.4% on CrossTask. It is because POC can accurately predict the actions in a test video while pseudo-label based methods [30, 37, 45] compute the similarity between the video and each training transcript with Viterbi decoding and consider the most similar one as the test transcript, which is often partially or fully incorrect. Learning and inference with Viterbi decoding also makes those methods time and computationally expensive. In contrast, our method is lightweight, learning actions with video-level losses and action ordering with the POC loss. *The average inference time of CDFL and TASL is 56 seconds per video, while POC only takes 0.014 seconds.* On the other hand, having ground-truth transcripts for action alignment, transcript-supervised methods achieve higher results as they separately estimate the temporal region for each occurrence of an action while this is not learned by modeling the pairwise action ordering via the POC Loss. Yet, POC exceeds CDFL at IoU on Breakfast and at both MoF and IoU on CrossTask, thanks to we end-to-endly train a deep network, enabled by differentiability of the POC loss, thus learn better action features.

	Segmentation		Alignment	
Breakfast	MoF	IoU	MoF	IoU
OCDC [2]	8.9			
CTC [18]	21.8			
HTK [23]	25.9	9.8	43.9	26.6
ECTC [18]	27.7		35.0	
HMM/RNN [43]	33.3			
TCFPN [9]	38.4	24.2	53.5	35.3
NNV [45]	43.0			
D3TW [5]	45.7		57.0	
CDFL [30]	**50.2**	33.7	63.0	45.8
MuCon [58]	48.5±1.8			
TASL [37]	49.9	36.6	**65.8**	**49.9**
POC (best)	47.1	**39.4**	56.1	46.7
POC (average)	45.7	38.3	54.4	46.4
CrossTask	MoF	IoU	MoF	IoU
NNV [45]	27.0	11.0	34.6	15.3
CDFL [30]	32.5	11.8	46.7	17.2
TASL [37]	42.7	14.9	**57.1**	**19.1**
POC (best)	**44.1**	**16.3**	53.3	18.9
POC (average)	42.8	15.6	53.0	18.4

Table 2. Transcript-supervised action segmentation and alignment results.

	Breakfast	CrossTask
action-set	0.264	0.412
transcript	0.147	0.332

Table 3. Ordering discrepancy between model predictions and ground-truth on test videos under different supervision.

Notice that using transcripts rather than action-sets greatly improves the performance of POC on Breakfast, showing the efficacy of POC loss in learning ordering. On CrossTask, the result for action segmentation is similar when using action-sets or transcripts, because videos have many 'background' segments between action segments, thus action ordering provides limited information about their locations. However, we show that POC uses the ground-truth transcripts to learn a better ordering. In Table 3, we report average order discrepancy defined in (6) between model predictions and ground-truth labels, i.e., reference ordering $O^\star$ is computed from the true labels. Using transcript supervision successfully reduces the discrepancy on both datasets.

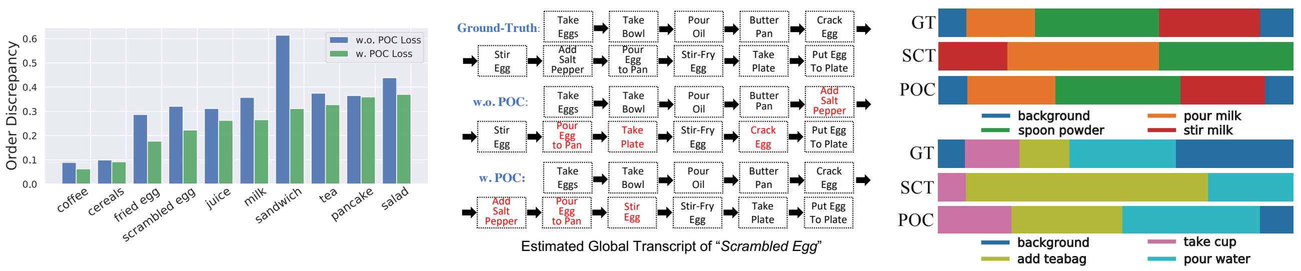


Figure 4. Left: ordering discrepancy between model predictions and ground-truth for each recipe in Breakfast. Middle: a global transcript (a single transcript for all videos) of the task *scrambled egg* estimated from transcripts of videos and learned by action ordering (red shows actions with incorrect positions). Right: action segmentation results by SCT and POC against ground-truth on Breakfast.

$\mathcal{L}_{\text{v-rk}}$	$\mathcal{L}_{\text{v-ce}}$	$\mathcal{L}_{\text{f-rk}}$	$\mathcal{L}_{\text{f-bk}}$	$\mathcal{L}_{\text{f-cr}}$	$\mathcal{L}_{\text{poc}}$	MoF	IoU
✓	✓	×	×	×	×	15.5	9.3
✓	✓	✓	×	×	×	36.6	28.7
✓	✓	✓	✓	×	×	36.8	29.4
✓	✓	✓	✓	×	✓	38.5	31.7
✓	✓	✓	✓	✓	×	37.0	30.3
✓	×	✓	✓	✓	✓	32.5	20.8
×	○	✓	✓	✓	✓	32.7	25.2
✓	✓	✓	✓	✓	✓	40.1	32.5

Table 4. Effect of proposed losses for action segmentation on Breakfast.

Ablation Studies. In Table 4, we summarize the contribution of each of our proposed loss functions by showing the average performance over three runs. The first row shows the baseline model with only video-level losses while last row is the complete model with all losses. First, adding $\mathcal{L}_{\text{f-rk}}$ improves the baseline by a factor of 2 and 3 on, respectively, MoF and IoU. Notice that with $\mathcal{L}_{\text{f-rk}}$, our model already outperforms ACV and SCT for action segmentation. Also, comparing the fourth and the last row shows that removing $\mathcal{L}_{\text{f-cr}}$ reduces MoF and IoU by 1.6% and 0.8% respectively. Comparing the fifth and last row shows removing $\mathcal{L}_{\text{poc}}$ reduces MoF and IoU by 3.1% and 2.2% respectively. Finally, as the fifth and sixth rows show, we test learning action recognition with only the ranking loss $\mathcal{L}_{\text{rank}}$ or the standard cross entropy loss without threshold estimation, denoted by ○. In both cases, MoF drops by 7%.

Effect of POC Loss. First, to understand the impact of POC loss under different scenarios, we show the ordering discrepancy defined in (6) between the ground-truth labels and our model predictions, i.e., $O^\star$ is computed from true labels in Figure 4 (left). We compare our model learned without POC loss (blue) and with POC loss (green) for each task (recipe) in Breakfast. Notice that the green bar is lower than blue bar for all recipes, including hard recipes like *fried egg*, *scrambled egg* and especially *sandwich*. The videos of the tasks are long videos with many actions. Both blue and green bars are high on *salad* and *pancake*, where actions often occur multiple times in a video, showing repeated action is a major challenge in learning action ordering.

To visualize the difference in the learned action ordering, in Figure 4 (middle), we show the global transcript (a single transcript for all videos) of the recipe *scrambled egg* estimated from our model without and with the POC loss.The global transcript is estimated via Bradley-Terry model [3] thanks to our ordering score $O(a_u, a_v)$ in (4), which can be viewed as the probability of a_u being before a_v. Without POC loss, 4 actions in the global transcript have incorrect positions, especially for *take plate* and *crack egg*, resulting in 7 wrong-ordered action pairs. In contrast, with the POC loss, there are only 2 incorrect ordering, between (*stir egg*, *add salt pepper*) and (*stir egg*, *pour egg to pan*).

Qualitative Results. In Figure 4 (right), we visualize the action segmentation results by SCT and POC against the ground-truth (GT). Notice that for the top example, SCT predicts a wrong ordering between *stir milk* and other actions while POC learns the correct ordering, thus, significantly improves the segmentation result. The bottom example shows a failure case of the set-supervised methods where SCT and POC predict the correct actions and ordering, yet the localizations do not align with the ground-truth. However, compared with SCT, POC predicts more accurate action boundaries between non-background actions.

5. Conclusions

We addressed the problem of set-supervised action learning in procedural task videos. We proposed an attention-based method with a novel differentiable pairwise ordering consistency loss that enforces similar action ordering for videos of the same task, resolving the drawbacks of conventional sequence alignment methods. Also, our method extends prior attention-based action localization networks to learn multiple actions in a video. We showed by experiments on three datasets that our method outperformed prior set-supervised algorithms and could be effectively extended to transcript supervision with minor modification.

Acknowledgements

This work is sponsored by DARPA PTG (HR00112220001), NSF (IIS-2115110), ARO (W911NF2110276) and ONR (N000141812132). Content does not necessarily reflect the position/policy of the Government. No official endorsement should be inferred.

References

[1] J. B. Alayrac, P. Bojanowski, N. Agrawal, J. Sivic, I. Laptev, and S. Lacoste-Julien. Unsupervised learning from narrated instruction videos. *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[2] P. Bojanowski, R. Lajugie, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic. Weakly supervised action labeling in videos under ordering constraints. *European Conference on Computer Vision*, 2014. 2, 7

[3] Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 1952. 8

[4] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 6

[5] Chien-Yi Chang, De-An Huang, Yanan Sui, Li Fei-Fei, and Juan Carlos Niebles. D3tw: Discriminative differentiable dynamic time warping for weakly supervised action alignment and segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1, 2, 3, 7

[6] M. Cuturi and M. Blondel. Soft-dtw: a differentiable loss function for time-series. *International Conference on Machine learning*, 2017. 3

[7] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[8] Fred J Damerau. A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 1964. 3

[9] Li Ding and Chenliang Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2, 7

[10] Nikita Dvornik, Isma Hadji, Konstantinos G Derpanis, Animesh Garg, and Allan D Jepson. Drop-dtw: Aligning common signal between sequences while dropping outliers. *Neural Information Processing Systems*, 2021. 3, 5

[11] E. Elhamifar and D. Huynh. Self-supervised multi-task procedure learning from instructional videos. *European Conference on Computer Vision*, 2020. 2

[12] E. Elhamifar and Z. Naing. Unsupervised procedure learning via joint dynamic summarization. *International Conference on Computer Vision*, 2019. 2

[13] Mohsen Fayyaz and Jurgen Gall. Sct: Set constrained temporal transformer for set supervised action segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2, 6, 7

[14] D. F. Fouhey, W. C. Kuo, A. A. Efros, and J. Malik. From lifestyle vlogs to everyday interactions. *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[15] Daniel Fried, Jean-Baptiste Alayrac, Phil Blunsom, Chris Dyer, Stephen Clark, and Aida Nematzadeh. Learning to segment actions from observation and narration. *Annual Meeting of the Association for Computational Linguistics*, 2020. 2

[16] Karan Goel and Emma Brunskill. Learning procedural abstractions and evaluating discrete latent temporal structure. *International Conference on Learning Representation*, 2019. 2

[17] Isma Hadji, Konstantinos G. Derpanis, and Allan D. Jepson. Representation learning via global temporal alignment and cycle-consistency. *IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 3

[18] D. A. Huang, L. Fei-Fei, and J. C. Niebles. Connectionist temporal modeling for weakly supervised action labeling. *European Conference on Computer Vision*, 2016. 2, 7

[19] Satoshi Koide, Keisuke Kawano, and Takuro Kutsuna. Neural edit operations for biological sequences. *Advances in Neural Information Processing Systems*, 2018. 3

[20] O. Koller, H. Ney, and R. Bowden. Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled. *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[21] O. Koller, S. Zargaran, and H. Ney. Re-sign: Re-aligned end-to-end sequence modeling with deep recurrent cnn-hmms. *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[22] H. Kuehne, A. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human. *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 4, 6

[23] H. Kuehne, J. Gall, and T. Serre. An end-to-end generative framework for video segmentation and recognition. *IEEE Winter Conference on Applications of Computer Vision*, 2016. 1, 7

[24] H. Kuehne, A. Richard, and J. Gall. Weakly supervised learning of actions from transcripts. *Computer Vision and Image Understanding Journal*, 2017. 2

[25] Anna Kukleva, Hilde Kuehne, Fadime Sener, and Jurgen Gall. Unsupervised learning of action classes with continuous temporal embedding. *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2, 5

[26] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager. Temporal convolutional networks for action segmentation and detection. *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1

[27] Pilhyeon Lee, Youngjung Uh, and Hyeran Byun. Background suppression network for weakly-supervised temporal action localization. In *AAAI*, 2020. 2

[28] Pilhyeon Lee, Jinglu Wang, Yan Lu, and Hyeran Byun. Weakly-supervised temporal action localization by uncertainty modeling. In *AAAI*, 2021. 2, 6, 7

[29] V. I. Levenshtein. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10, 1966. 2, 3

[30] J. Li, P. Lei, and S. Todorovic. Weakly supervised energy-based learning for action segmentation. *International Conference on Computer Vision*, 2019. 1, 2, 6, 7

[31] Jun Li and Sinisa Todorovic. Set-constrained viterbi for set-supervised action segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2, 4, 6, 7

[32] J. Li and S. Todorovic. Action shuffle alternating learning for unsupervised action segmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 2

[33] J. Li and S. Todorovic. Anchor-constrained viterbi for set-supervised action segmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1, 2, 4, 6, 7

[34] Y. Li, Y. Song, and J. Luo. Improving pairwise ranking for multi-label image classification. *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 5

[35] Zhe Li, Yazan Abu Farha, and Juergen Gall. Temporal action segmentation from timestamp supervision. *ICCV*, 2021. 3

[36] M. Lin, N. Inoue, and K. Shinoda. Ctc network with statistical language modeling for action sequence recognition in videos. *Thematic Workshops of the ACM Conference on Multimedia*, 2017. 2

[37] Z. Lu and E. Elhamifar. Weakly-supervised action segmentation and alignment via transcript-aware union-of-subspaces learning. *International Conference on Computer Vision*, 2021. 1, 2, 6, 7

[38] Wang Luo, Tianzhu Zhang, Wenfei Yang, Jingen Liu, Tao Mei, Feng Wu, and Yongdong Zhang. Action unit memory network for weakly supervised temporal action localization. In *CVPR*, 2021. 2

[39] Zhekun Luo, Devin Guillory, Baifeng Shi, Wei Ke, Fang Wan, Trevor Darrell, and Huijuan Xu. Weakly-supervised action localization with expectation-maximization multi-instance learning. In *ECCV 2020*, 2020. 2

[40] Junwei Ma, Satya Krishna Gorti, Maksims Volkovs, and Guangwei Yu. Weakly supervised action selection learning in video. In *CVPR*, 2021. 2

[41] A. Miech, D. Zhukov, J. B. Alayrac, M. Tapaswi, I. Laptev, and J. Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. *International Conference on Computer Vision*, 2019. 2

[42] Z. Naing and E. Elhamifar. Procedure completion by learning from partial summaries. *British Machine Vision Conference*, 2020. 2

[43] A. Richard, H. Kuehne, and J. Gall. Weakly supervised action learning with rnn based fine-to-coarse modeling. *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 7

[44] A. Richard, H. Kuehne, and J. Gall. Action sets: Weakly supervised action segmentation without ordering constraints. *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2, 4, 6, 7

[45] A. Richard, H. Kuehne, A. Iqbal, and J. Gall. Neuralnetwork-viterbi: A framework for weakly supervised video learning. *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2, 6, 7

[46] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 1

[47] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer Vision*, 2015. 4, 6

[48] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26, 1978. 2, 3

[49] Fadime Sener and Angela Yao. Unsupervised learning and segmentation of complex activities from video. *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[50] Ozan Sener, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. Unsupervised semantic parsing of video collections. *IEEE International Conference on Computer Vision*, 2015. 2

[51] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Intra- and inter-action understanding via temporal action parsing. *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2

[52] Y. Shen and E. Elhamifar. Semi-weakly-supervised learning of complex actions from instructional task videos. *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 1, 3

[53] Y. Shen, L. Wang, and E. Elhamifar. Learning to segment actions from visual and language instructions via differentiable weak sequence alignment. *IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 2, 3, 5

[54] Baifeng Shi, Qi Dai, Yadong Mu, and Jingdong Wang. Weakly-supervised action localization by generative attention modeling. *CVPR*, 2020. 2

[55] G. A. Sigurdsson, S. Divvala, A. Farhadi, and A. Gupta. Asynchronous temporal fields for action recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1

[56] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. *European Conference on Computer Vision*, 2016. 2

[57] B. Singh, T. K. Marks, M. Jones, O. Tuzel, and M. Shao. A multi-stream bi-directional recurrent neural network for finegrained action detection. *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1

[58] Yaser Souri, Mohsen Fayyaz, Luca Minciullo, Gianpiero Francesca, and Juergen Gall. Fast Weakly Supervised Action Segmentation Using Mutual Consistency. *PAMI*, 2021. 1, 2, 6, 7

[59] Yaser Souri, Alexander Richard, Luca Minciullo, and Juergen Gall. On evaluating weakly supervised action segmentation methods, 2020. 6

[60] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alexander Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *Arxiv*, 2016. 3

[61] C. Xu and E. Elhamifar. Deep supervised summarization: Algorithm and application to learning instructions. *Neural Information Processing Systems*, 2019. 2

[62] S. Yeung, O. Russakovsky, N. Jin, M. Andriluka, G. Mori, and L. Fei-Fei. Every moment counts: Dense detailed labeling of actions in complex videos. *International Journal of Computer Vision*, 2018. 1

[63] Can Zhang, Meng Cao, Dongming Yang, Jie Chen, and Yuexian Zou. Cola: Weakly-supervised temporal action localization with snippet contrastive learning. In *CVPR*, 2021. 2

[64] D. Zhukov, J. B. Alayrac, R. G. Cinbis, D. Fouhey, I. Laptev, and J. Sivic. Cross-task weakly supervised learning from instructional videos. *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2, 4, 6