This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Understanding Multi-Task Activities from Single-Task Videos

Yuhan Shen
Northeastern University
shen.yuh@northeastern.edu

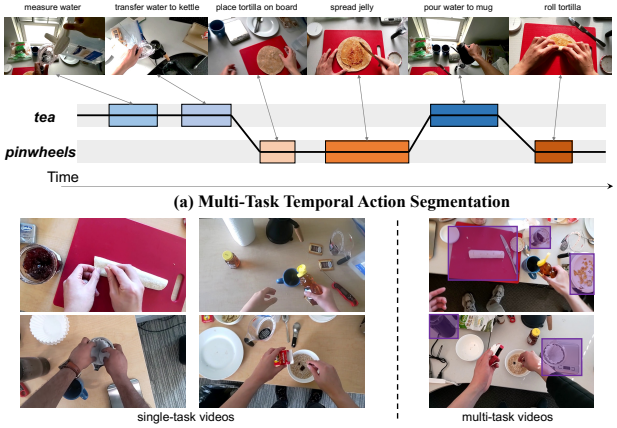
Ehsan Elhamifar
Northeastern University
e.elhamifar@northeastern.edu

Abstract

We introduce and develop a framework for Multi-Task Temporal Action Segmentation (MT-TAS), a novel paradigm that addresses the challenges of interleaved actions when performing multiple tasks simultaneously. Traditional action segmentation models, trained on single-task videos, struggle to handle task switches and complex scenes inherent in multi-task scenarios. To overcome these challenges, our MT-TAS approach synthesizes multi-task video data from single-task sources using our Multi-task Sequence Blending and Segment Boundary Learning modules. Additionally, we propose to dynamically isolate foreground and background elements within video frames, addressing the intricacies of object layouts in multi-task scenarios and enabling a new two-stage temporal action segmentation framework with Foreground-Aware Action Refinement. Also, we introduce the Multi-task Egocentric Kitchen Activities (MEKA) dataset, containing 12 hours of egocentric multi-task videos, to rigorously benchmark MT-TAS models. Extensive experiments demonstrate that our framework effectively bridges the gap between single-task training and multi-task testing, advancing temporal action segmentation with state-of-the-art performance in complex environments.[1]

1. Introduction

Consider a typical morning routine, when you prepare breakfast: you start making oatmeal and while waiting for it to cook, you brew coffee and pack your food for lunch. In fact, in our daily lives, we often perform multiple tasks simultaneously. For computer vision systems to accurately analyze such behaviors and provide useful assistance, they must comprehend these complex and interleaved activities. Temporal action segmentation (TAS) has emerged as a crucial technology for understanding human activities by partitioning videos into meaningful segments, each corresponding to a distinct action or step [17, 22, 24, 31, 33, 48, 49, 64, 73, 84, 88]. Recently, TAS has attracted growing interests within egocentric videos [5, 36, 52, 54, 65, 77], where activities are captured from a first-person perspective, enabling fine-grained analysis of human actions, which is essential for applications such as real-time personalized assistance, embodied agents and efficient video content retrieval.



(a) Multi-Task Temporal Action Segmentation

(b) Comparison between frames from single-task videos *v.s.* multi-task videos

Figure 1. (a) Multi-task temporal action segmentation involves segmenting actions from interleaved tasks. It requires capturing interruptions and resumptions in the execution of each task. (b) Comparison between frames from single-task and multi-task videos. In multi-task videos, at each time instant, the scene often contains objects from tasks different from the currently executed one (highlighted by purple boxes).

In this paper, we study the new problem of *Multi-Task Temporal Action Segmentation* (MT-TAS), which explicitly focuses on scenarios where users perform multiple tasks by interleaving their steps within a single video, see Fig. 1. There are *two major challenges* that existing TAS models face in addressing MT-TAS. First, despite the notable progress through deep learning techniques [1, 6, 14, 17, 41, 42, 45, 49, 84], current TAS models have primarily focused on scenarios where each video contains actions related to a single high-level task, such as assembling a piece of furniture or cooking a specific recipe. This single-task focus limits the applicability of TAS models to the multi-task setting since i) they are biased to predict actions/steps from only one task at the inference time, ii) when performing several tasks, the environment and the scene often contains objects relevant to multiple tasks, which adds complexity

[1]Code: https://github.com/Yuhan-Shen/MT-TAS.

to the context, while single-task videos often contain objects relevant to one task (see Fig. 1(b)), iii) interleaving steps of multiple tasks introduces complexities such as unpredictable task switches and disruptions to the temporal flows of actions within each task.

Second, a significant obstacle in advancing MT-TAS is the lack of appropriate training data. Existing datasets [5, 16, 18, 31, 36, 63, 88] are predominantly collected for single-task scenarios, making them insufficient for capturing complexities of interleaved tasks. Furthermore, many possible step combinations in the multi-task setting makes it impractical to collect and annotate comprehensive real-world videos covering all potential situations (such videos would also be much longer than single-task videos).

Paper Contributions. We study the new problem of MT-TAS. To overcome the aforementioned challenges, we propose an MT-TAS framework that uses only single-task videos for training. We generate synthetic multi-task data using single-task videos and propose a two-stage action segmentation model for accurate predictions. For the data generation, we first develop a *Multi-task Sequence Blending* (MSB) module that leverages Large Language Models (LLMs) to create multi-task transcripts of interleaved steps and their associated videos. Second, we develop a *Segment Boundary Learning* (SBL) module to mitigate the abrupt transitions at segment boundaries caused by the synthetic data. Third, we introduce a *Dynamic Isolation of Video Elements* (DIVE) approach to isolate foreground and background components within video frames by identifying action-relevant objects and combining foregrounds of actions in one task with backgrounds of other tasks to capture the more complex scenes of multi-task videos. By increasing the variation in the background components for the training data, our method becomes more robust to unseen object compositions in multi-task scenarios.

For the action segmentation model, we develop a two-stage framework, where we refine the initial action predictions using a Foreground-Aware Action Refinement (FAAR) module. This approach leverages foreground information to enhance the focus of our model on action-relevant cues (e.g., focusing on the regions of kettle and measuring cup for the action "transfer water to kettle" as in Fig. 3(b)), improving accuracy and robustness to background variations during the multi-task testing.

For proper evaluation of MT-TAS models, we gather and annotate the *Multi-task Egocentric Kitchen Activities* (MEKA) video dataset (we plan to publicly release the dataset), built upon the recent EgoPER [36] dataset. MEKA comprises 100 egocentric videos of users (about 12 hours of footage) performing multiple kitchen recipes from EgoPER in each run by interleaving their steps. This new benchmark captures a variety of interleaved kitchen activities, providing a rigorous testing ground for MT-TAS models. Through extensive experiments in both offline and online settings, we demonstrate that our proposed framework successfully bridges the gap between single-task training data and the complexities of real-world multi-task scenarios, advancing research in temporal action segmentation.

2. Related Works

Temporal Action Segmentation. TAS aims to understand the sequential structure of human activities in videos by assigning an action label to each frame, thereby partitioning the video into a sequence of action segments. Based on the level of supervision during training, TAS methods can be categorized into three main groups: unsupervised [2, 15, 16, 20, 33, 34, 61, 66, 76, 79, 80, 88], weakly-supervised [8, 12, 19, 22–24, 38–40, 44, 47, 48, 57–59, 62, 64, 71], and fully-supervised [1, 4, 6, 17, 27, 29, 32, 35, 36, 41, 42, 45, 49, 51, 60, 65, 68–71, 83, 84] approaches. Our work primarily aligns with the fully-supervised methods. However, regardless of the supervision level, previous studies have predominantly focused on *single-task* TAS, where test videos contain actions from only one high-level activity or task, while we focus on *multi-task* TAS. A closely related work, UnweaveNet [53], aims to predict the start, continuation, and resumption of multiple activity threads within a video. While UnweaveNet models multiple concurrent activities, its predictions are at the activity or task level and do not capture the fine-grained action steps within each task. In contrast, our approach focuses on recognizing and segmenting detailed action steps across interleaved tasks, providing a more granular understanding of multi-task activities.

Procedural Video Datasets. To support research in TAS and related tasks, numerous procedural video datasets have been collected and released. These datasets include third-person videos (*e.g.*, Breakfast [30], 50Salads [72], YouTube Instructional [2], COIN [74], CrossTask [88], ProceL [16], and ATA [24]), egocentric videos (*e.g.*, GTEA [18], EGTEA [43], MECCANO [56], EgoProceL [5], HoloAssist [78], and EgoPER [36]), or both views such as Assembly101 [63], (3+1)ReC [62] and EgoExoLearn [28]. While large-scale video datasets like EPIC-Kitchens [11], Ego4D [25], and Ego-Exo4D [26] exist, they primarily focus on detecting and recognizing atomic action steps rather than modeling the procedural relationships between steps within a task, making them less suitable for TAS research. Also, most existing TAS datasets are limited to single-task scenarios, where each video contains actions related to one high-level activity. In contrast, our collected MEKA dataset specifically addresses multi-task temporal action segmentation by capturing videos where multiple tasks are performed in an interleaved manner, providing a more challenging and realistic benchmark for advancing TAS research.

Object-Aware and Foreground/Background Learning.

Learning to localize active objects or relevant objects is a prominent topic in video understanding [67, 75, 85, 86]. Furthermore, researchers have explored leveraging object information to improve video representation learning [85], action recognition [87], and action anticipation [86]. Some studies have also focused on disentangling foreground and background elements to enhance video representations [3, 13, 37]. Our work differs by specifically addressing the generalization of TAS models to multi-task videos. Unlike existing approaches that concentrate on short video clips or single-task videos, we propose a method that dynamically isolates foreground and background elements to handle unseen object compositions in multi-task scenarios, thereby mitigating interference from irrelevant objects.

3. Multi-Task Temporal Action Segmentation

3.1. Problem Setting

Given a video sequence $\mathcal{V} = (v_1, \ldots, v_T)$ of T frames, the goal of TAS is to predict the sequence of action labels $A = (a_1, \ldots, a_T)$, where each label a_t comes from a predefined set of action classes $\mathcal{C}$. Assume we have O tasks, where the set $\mathcal{C}_o$ consists of actions for the task $o \in \{1, \ldots, O\}$ and $\mathcal{C} = \bigcup_{o=1}^{O} \mathcal{C}_o$ contains actions of all tasks. In a *single-task* video, actions come from one $\mathcal{C}_o$, while a *multi-task* video has interleaved actions from several tasks, hence, a_t may belong to any $\mathcal{C}_o$ for $o \in \{1, \ldots, O\}$. Additionally, there may be *background* segments (e.g., 'answering phone') irrelevant to any of the tasks. A particular challenge in MT-TAS is to distinguish between action segments belonging to different tasks and background segments.

In our setting, the training data consists of single-task videos with full framewise annotations. Our goal is to train a model on single-task videos so that it can effectively generalize to multi-task videos during testing.

3.2. Overview of Proposed Framework

Fig. 2 shows the overview of our MS-TAS framework. Our approach first employs Multi-task Sequence Blending (MSB) to generate synthetic multi-task videos by leveraging LLM-guided task transitions, eliminating the need for multi-task data collection. However, the initial synthesized multi-task videos contain abrupt transitions at segment boundaries, which disrupt the temporal flow, and contain discrepancies in object compositions between single-task and multi-task videos. To mitigate these issues, we introduce three components: a Segment Boundary Learning (SBL) module that enhances temporal smoothness between segments, a Dynamic Isolation of Video Elements approach that augments the background variations during training, and a two-stage TAS method with foreground-aware action refinement to improve the predictions of the model. Next, we discuss each component of our framework in details.

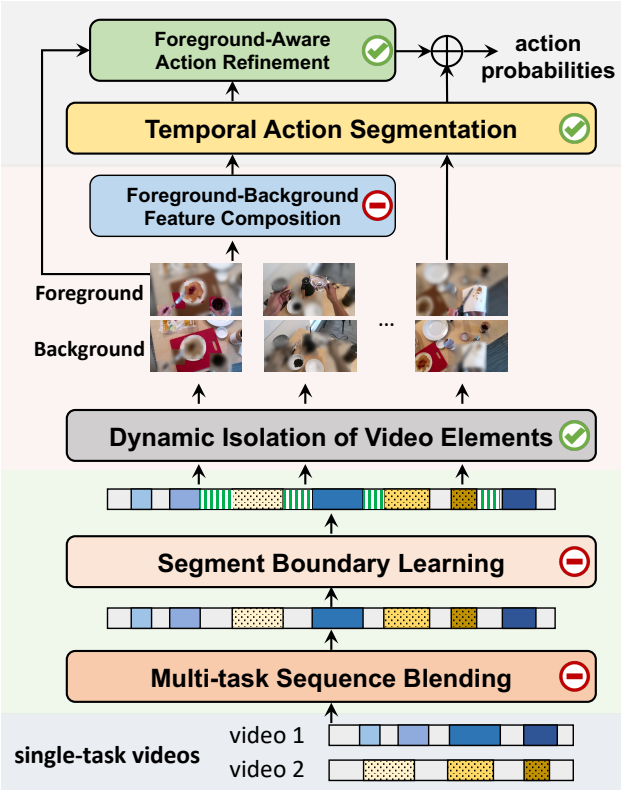
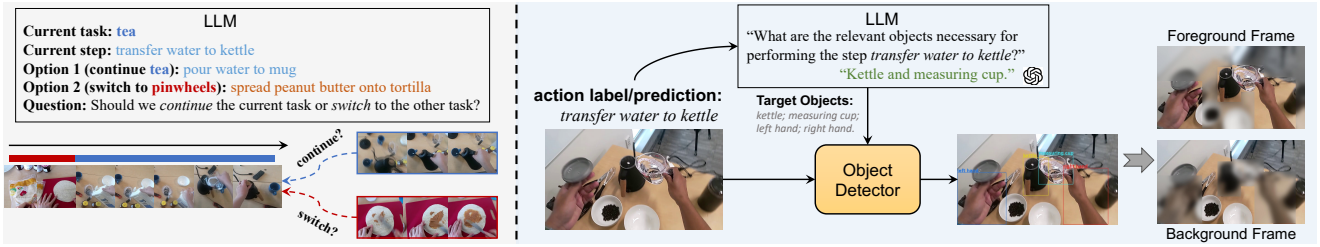


Figure 2. Framework Overview. The modules marked with ⊖ are used exclusively for training, and modules marked with ✓ are used for both training and testing.

3.3. Multi-task Sequence Blending (MSB)

The goal of MSB is to create initial multi-task video sequences from single-task videos, to simulate real-world multitasking behaviors. Given single-task training videos $\{\mathcal{V}^{(i)}\}_{i=1}^{M}$, where each video $\mathcal{V}^{(i)} = (v_1^{(i)}, v_2^{(i)}, \ldots, v_{T_i}^{(i)})$ has a task label τ_i and consists of a sequence of action segments $S^{(i)} = (s_1^{(i)}, s_2^{(i)}, \ldots, s_{N_i}^{(i)})$ with action labels $A^{(i)} = (a_1^{(i)}, a_2^{(i)}, \ldots, a_{N_i}^{(i)})$, a naive approach would be to randomly sample segments from videos to concatenate. However, this ignores the temporal order of actions within a task and more importantly can create unrealistic sequences, e.g., "scooping jelly from jar" followed by "grinding coffee beans" (this is unrealistic since it is common sense to spread the jelly right away after scooping it from a jar).

To create plausible multi-task sequences, we: i) preserve the original temporal order of action segments within each task, ii) preserve common sense action ordering and switches among tasks. Specifically, the sequence of action segments $s_j^{(i)}$ from task τ_i follows the same order as in the original single-task video. At the end of an action segment, we leverage an LLM, which encodes extensive contextual and commonsense knowledge from real-world data, to decide whether to continue the current task or switch to an-



(a) Multi-task Sequence Blending

(b) Dynamic Isolation of Video Elements

Figure 3. Illustration of leveraging LLM in our proposed framework for MSB (a) and DIVE (b).

other task (see Fig. 3(a)). The LLM assesses the plausibility of continuing or switching based on the current action $a_j^{(i)}$, the next action of the current task $a_{j+1}^{(i)}$, and the next action of another task $a_m^{(k)}$, where τ_k is a different task and $a_m^{(k)}$ is its next pending action. For example, after "turning on the kettle to boil water" in a tea-making task, it is plausible to switch to a new task during the waiting time. Conversely, after "scooping jelly from a jar" in a pinwheel-making task, it makes sense to continue with the next action "spread jelly on tortilla" rather than switching tasks.

During training, we randomly sample potential task switch points at the end of action segments and use the LLM to decide whether to continue or switch tasks. For more than two tasks, we first randomly select one unfinished task as the candidate for switching. We then blend the sequence of action segments based on the LLM's decisions, resulting in a new multi-task sequence of actions and frames.

3.4. Segment Boundary Learning (SBL)

The multi-task videos synthesized by MSB exhibit abrupt transitions at boundaries where we switch tasks, which disrupt the temporal flow and negatively impact the model's performance. Therefore, we propose the SBL module that learns to recover the feature representation of a frame f_t using its nearby frames, excluding frames immediately adjacent to t to avoid trivial learning. This approach allows us to generate features for frames where we concatenate segments from different videos, thus enhancing the temporal smoothness between action segments.

Following prior works in TAS [17, 45, 49, 84], we first extract the framewise feature representation f_t using an I3D feature extractor [7] applied to a sliding window V_t,

$$f_t = \text{I3D}(V_t). \tag{1}$$

Then, for a given frame t, we consider its neighboring frames from the intervals $[t-\delta-w,\ t-\delta]$ and $[t+\delta,\ t+\delta+w]$, where w defines the window size and δ specifies the exclusion margin around t. We train a reconstruction function $\mathcal{F}_{\text{SBL}}$ to predict the feature f_t from its surrounding frames,

$$\bar{f}_t = \mathcal{F}_{\text{SBL}}\big(f_{t-\delta-w:t-\delta},\ f_{t+\delta:t+\delta+w}\big), \tag{2}$$

where $f_{t-\delta-w:t-\delta}$ and $f_{t+\delta:t+\delta+w}$ denote the features of frames before and after t within the specified intervals. We optimize the reconstruction function by minimizing the reconstruction loss over *non-boundary frames*,

$$\mathcal{L}_{\text{SBL}} = \sum_{t \notin \mathcal{B}} \left\| \bar{f}_t - f_t \right\|^2, \tag{3}$$

where $\mathcal{B}$ is the set of boundary frames (which come from two different videos, therefore are not continuous or realistic and we should not learn from them). By learning this reconstruction function, the model can generate smooth feature representations at segment boundaries. During training, we replace the original features at boundary frames with the reconstructed features $\bar{f}_t$ to enhance the temporal smoothness of the synthesized multi-task videos.

3.5. Dynamic Isolation of Video Elements (DIVE)

In real multi-task videos, the present objects and their layouts can significantly differ from those in single-task videos. For example, as shown in the top-right frame in Fig. 1(b), in a video of making both tea and pinwheels, we might add honey to a mug while a rolled tortilla is on the cutting board, which is typically unseen in single-task videos. However, the output of MSB, which concatenates different actions from different single-task videos, does not capture such realistic multi-task layouts. Observing that action-relevant objects remain consistent across single-task and multi-task videos, we propose a *Dynamic Isolation of Video Elements* approach to isolate video elements into foreground and background components, and develop a *Foreground-Background Feature Composition* module to synthesize more realistic multi-task videos for training.

Detecting Action-Relevant Objects. Given a frame, it is challenging to distinguish between foreground and background solely based on visual cues, especially in cluttered multi-task environments. The definition of "foreground" or "relevant objects" depends on the action being performed; thus, knowledge of the action context is essential. Fig. 3(b) illustrates how we obtain the foreground and background components. We first query an LLM to obtain a list of relevant objects for each action class $a \in \mathcal{C}$. For example,

the action "transfer water to kettle" would be associated with "kettle" and "measuring cup". Then, we use an open-vocabulary object detector, GroundingDINO [46], to detect these relevant objects along with hands (we add "left hand" and "right hand" as object names) within each frame. We define the foreground region as the union of detected bounding boxes. We generate foreground and background frames through Gaussian blurring [82] outside and inside this region, respectively. We empirically observed that Gaussian blurring performs slightly better than simply zero masking.

Foreground-Background Feature Composition (FBFC). After generating foreground and background frames, we extract I3D features,

$$f_t^{\text{fg}} = \text{I3D}(V_t \odot M_t^{\text{fg}}), \quad f_t^{\text{bg}} = \text{I3D}(V_t \odot M_t^{\text{bg}}), \tag{4}$$

where f_t^{fg} and f_t^{bg} are foreground and background frame features, respectively, and $\odot$ is the Gaussian blurring operation with corresponding foreground masks M_t^{fg} and background masks M_t^{fg}. We train a decoder $\mathcal{D}$ to reconstruct the original feature from foreground and background features,

$$\hat{f}_t = \mathcal{D}(f_t^{\text{fg}}, f_t^{\text{bg}}). \tag{5}$$

Let f_t denote the original feature in Eq. (1). We train the decoder by minimizing the composition loss,

$$\mathcal{L}_{\text{FBFC}} = \sum_t \left\| \hat{f}_t - f_t \right\|^2. \tag{6}$$

To augment synthetic multi-task features, we mix the background features of the current frame with background features of frames from another task

$$\tilde{f}_t^{\text{bg}} = \beta f_t^{\text{bg}} + (1-\beta) f_{t'}^{\text{bg}}, \tag{7}$$

where f_t^{bg} is the background feature from the current frame, $f_{t'}^{\text{bg}}$ is the background feature from another task, and $\beta \in [0, 1]$ controls the mixing ratio. We then compose multi-task features from the foreground feature and the mixed background feature using the decoder

$$\tilde{f}_t = \mathcal{D}\big(f_t^{\text{fg}}, \tilde{f}_t^{\text{bg}}\big). \tag{8}$$

We replace the training data with these recomposed features, exposing the model to diverse background variations.

3.6. Two-Stage Temporal Action Segmentation

To further enhance the robustness of our model to background variations during testing, we propose a two-stage action segmentation method incorporating a **Foreground-Aware Action Refinement** (FAAR) module. By focusing on foreground regions, FAAR mitigates the impact of background distractions and refines the initial TAS predictions.

During training, with ground-truth actions available, we obtain the action-relevant objects and detect the foreground regions. We extract CLIP features [55] from the foreground frames and apply a lightweight TAS model to predict foreground-aware action probabilities based solely on the foreground frames, p_t^{fg}, which are combined with the initial action predictions p_t output by the base TAS model,

$$p_t^{\text{fg}} = \mathcal{R}\big(g(v_t \odot m_t^{\text{fg}})\big), \quad p_t^{\text{final}} = (1-\alpha)\cdot p_t + \alpha \cdot p_t^{\text{fg}}, \tag{9}$$

where m_t^{fg} is the binary foreground mask, g denotes CLIP feature extractor, $\mathcal{R}$ denotes the lightweight TAS model, $\alpha \in [0, 1]$ controls the weight of foreground-aware probabilities, and p_t^{final} is the refined action probabilities.

During inference, without ground-truth actions, we first select the top K predicted action classes $\{\hat{a}_t^{(1)}, \ldots, \hat{a}_t^{(K)}\}$ based on the initial action probabilities p_t from the base TAS model, and retrieve their relevant objects. We then detect the corresponding foreground regions and obtain foreground-aware action probabilities $p_t^{\text{fg},(k)}$ for each of the top K actions respectively. We compute the final prediction by combining the initial action probabilities with the foreground-aware ones,

$$p_t^{\text{final}} = (1-\alpha)\cdot p_t + \alpha \cdot \sum_{k=1}^{K} \theta_t^{(k)}\, p_t^{\text{fg},(k)}, \tag{10}$$

where the weights $\theta_t^{(k)}$ are proportional to the initial probabilities of the top-K actions, i.e., $\theta_t^{(k)} = \frac{p_t\left(\hat{a}_t^{(k)}\right)}{\sum_{j=1}^{K} p_t\left(\hat{a}_t^{(j)}\right)}$. This approach refines the predictions by focusing on the most probable actions and their relevant objects, enhancing robustness to background variations in multi-task scenarios.

3.7. Training and Inference

Training. We use the synthetic multi-task videos generated by MSB for training. We train the SBL module $\mathcal{F}_{\text{SBL}}$ via $\mathcal{L}_{\text{SBL}}$ in Eq. (3) and utilize the reconstructed features at segment boundaries to enhance temporal smoothness. Similarly, we train the FBFC decoder $\mathcal{D}$ via $\mathcal{L}_{\text{FBFC}}$ in Eq. (6), and augment the training data by mixing background features from different tasks as described in Eqs. (7) and (8). We train the base TAS model with FAAR using the standard action loss. The total loss function $\mathcal{L}$ is:

$$\mathcal{L} = \mathcal{L}_{\text{action}} + \lambda_{\text{SBL}}\mathcal{L}_{\text{SBL}} + \lambda_{\text{FBFC}}\mathcal{L}_{\text{FBFC}}, \tag{11}$$

where $\mathcal{L}_{\text{action}}$ includes the cross-entropy loss and smoothing loss for TAS, applied to the final action probabilities output by the FAAR module. λ_{SBL} and λ_{FBFC} are the loss weights.

Inference. During inference, we do not use the MSB, SBL and FBFC modules, as they are designed to augment training. We first obtain initial action predictions from the base TAS model, and then apply the FAAR module, as specified in Eq. (10), to derive the final action predictions.

3.8. Incorporating Unlabeled Multi-Task Videos

Our proposed framework so far trains an MT-TAS model using only single-task videos. However, we additionally investigate the potential benefits of incorporating *unlabeled* multi-task videos, when such videos are available. Since annotating long multi-task videos is labor-intensive and time-consuming, leveraging unlabeled multi-task videos offers a practical way to enhance the model performance.

We formulate this as a domain adaptation problem [9, 10, 50], where labeled single-task and synthetic multi-task videos constitute the *source domain*, while unlabeled real multi-task videos form the *target domain*. We aim to learn domain-invariant feature representations through adversarial training, achieved by adding a domain classifier with a Gradient Reversal Layer (GRL) [21] after the base TAS model's early feature layer. The model is trained jointly with action labels on the source domain and domain labels on both domains. During training, the GRL reverses the gradient flow, encouraging indistinguishable feature distributions across domains. Our overall training loss is

$$\mathcal{L} = \mathcal{L}_{\text{action}} + \lambda_{\text{SBL}}\mathcal{L}_{\text{SBL}} + \lambda_{\text{FBFC}}\mathcal{L}_{\text{FBFC}} + \lambda_{\text{domain}}\mathcal{L}_{\text{domain}}, \quad (12)$$

where $\mathcal{L}_{\text{domain}}$ is the framewise binary domain classification loss, and λ_{domain} is the loss weight.

4. MEKA Dataset

While our proposed framework can be trained using only single-task videos, we require an MT-TAS benchmark for evaluation. To this end, we introduce the **Multi-task Egocentric Kitchen Activities (MEKA)** dataset, which extends the EgoPER dataset [36] to multi-task scenarios. EgoPER is an egocentric procedural video dataset designed for temporal action segmentation and error detection. It contains single-task videos of users performing kitchen activities while wearing an egocentric camera (*HoloLens 2*). The dataset includes five cooking tasks: making *pinwheels*, *quesadillas*, *oatmeal*, *coffee*, and *tea*, each associated with a predefined task graph.

To create realistic multi-task scenarios, we generated multi-task transcripts by interleaving action sequences from multiple tasks while ensuring logical coherence within the task graphs. Participants equipped with a HoloLens 2 camera followed these transcripts during recording. While we used similar recording equipment and kitchen tools as in the EgoPER dataset to maintain consistency, variations in the environments introduce additional challenges to the evaluation benchmark.

After data collection, we annotated each video by assigning framewise labels for both actions and tasks. In total, MEKA consists of 100 videos for 12 hours of footage. In each video, participants perform two or three cooking tasks in an interleaved manner. See the supplementary materials for more details about the dataset. We will make the MEKA dataset publicly available upon publication, including videos and framewise annotations, to facilitate further research in multi-task temporal action segmentation.

5. Experiments

5.1. Experimental Setup

Datasets. Since MT-TAS is a new task, existing datasets cannot be used for its evaluation. Consequently, we evaluate our approach using single-task videos from EgoPER [36] for training and our collected multi-task videos from MEKA for testing. EgoPER contains 213 normal single-task egocentric videos and MEKA contains 100 multi-task videos for the five recipes in EgoPER. In total, it has 51 classes of action steps. We were also able to adapt the EGTEA dataset [43] for multi-task evaluation. We manually select actions that appear exclusively in either the first or second half of each video, treating these segments as single-task videos, and consider the full videos with interleaved actions as multi-task videos. The supplementary materials provide additional details and the experimental results on the adapted EGTEA dataset.

Model Architecture. We evaluate our method with three base TAS models: two offline models (MSTCN [17] and FACT [49]) and one online model (ProTAS [65]). Online TAS differs from offline TAS in that it only uses the frames up to the current moment for prediction. We implement the SBL module ($\mathcal{F}_{\text{SBL}}$) by processing interval features through two convolutional layers, followed by temporal global average pooling. The Decoder $\mathcal{D}$ processes foreground and background features through parallel two-layer MLPs, each reducing feature dimension by half, followed by a linear layer that reconstructs the original feature representation from the concatenated features. We employ a three-layer TCN [17] with a hidden dimension of 64 for FAAR ($\mathcal{R}$). For experiments with domain adaptation, we insert a GRL after the feature layer of the initial stage of TAS models, and use a two-layer MLP for binary domain classification.

Implementation Details. Following standard practice [17, 49, 65, 81, 84], we extract 2048-dimensional I3D features [7] pretrained on Kinetics [7] as video representations. We evaluate performance using standard TAS metrics: frame-wise accuracy (Acc), frame-wise accuracy excluding background frames (Acc-bg), segment-wise edit score (Edit), and segment-wise F1 scores at multiple IoU thresholds (F1@10,25,50). We adopt the default configurations from MSTCN [17], FACT [49], and ProTAS [65] using their official source codes. We train all models using Adam optimizer with a learning rate of 0.0005. We set the loss weights as $\lambda_{\text{SBL}} = \lambda_{\text{FBFC}} = \lambda_{\text{domain}} = 1$. For the SBL module, we define boundary frames as the 10 frames centered at the points where segments from different videos are concatenated, and set window size w as 5 and margin

MSB	SBL	FBFC	FAAR	TAS model: MSTCN [17] Acc	Acc-bg	Edit	F1@{10,25,50}			TAS model: FACT [49] Acc	Acc-bg	Edit	F1@{10,25,50}		
baseline				62.3	49.1	50.1	52.6	49.3	39.7	53.7	41.4	42.9	44.6	41.6	31.7
✓				67.8	63.7	64.2	68.9	66.8	56.0	72.8	67.8	73.1	73.1	70.0	58.4
✓	✓			68.8	65.9	64.7	70.8	68.8	58.1	73.6	70.4	74.5	76.1	73.6	62.3
✓	✓	✓		73.7	64.4	69.5	73.0	69.9	58.8	73.8	71.3	74.4	79.6	76.7	65.5
✓	✓	✓	✓	**75.7**	**72.1**	**74.9**	**79.7**	**77.6**	**67.4**	**75.7**	**72.8**	**76.0**	**81.2**	**79.3**	**69.9**

Table 1. Offline multi-task temporal action segmentation performance on MEKA.

MSB	SBL	FBFC	FAAR	Acc	Edit	F1@{10,25,50}		
baseline				49.1	24.6	24.1	20.0	12.4
✓				51.1	30.7	26.5	22.1	12.8
✓	✓			51.1	33.1	29.3	24.5	14.6
✓	✓	✓		55.2	37.5	32.8	27.1	18.3
✓	✓	✓	✓	**67.8**	**54.8**	**54.6**	**49.8**	**36.7**

Table 2. Online temporal action segmentation on MEKA.

δ as 2. For Foreground-Background Feature Composition (FBFC), we select 10% of frames and augment them with Eq. (7) by randomly sampling β from [0.5, 1] during training. See Sec. 5.2 for the qualitative analysis on the value of β. For the FAAR module, we employ a two-stage training process. In the first stage, we train the model without FAAR by setting $\alpha = 0$ in Eq. (9). In the second stage, we incorporate FAAR and randomly sample α from [0, 1] during training. For inference, we set $K = 3$ and $\alpha = 0.3$ in Eq. (10). We include ablation studies on the values of K and α in Sec. 5.2. Please refer to the supplementary materials for more implementation details.

5.2. Experimental Results

Offline Temporal Action Segmentation. Tab. 1 presents the performance of our approach on the MEKA dataset using two baseline offline TAS models: MSTCN [17] and FACT [49]. We start with the baseline models trained on single-task videos without any of our proposed modules (first row). We then progressively incorporate the Multi-task Sequence Blending (MSB), Segment Boundary Learning (SBL), Foreground-Background Feature Composition (FBFC), and Foreground-Aware Action Refinement (FAAR) modules to assess their individual contributions. In the first row, MSTCN achieves an accuracy of 62.3% and an F1@50 score of 39.7%, while FACT attains 53.7% accuracy and 31.7% F1@50, indicating that the baseline models struggle to generalize to multi-task videos when trained solely on single-task data. Incorporating MSB significantly improves performance for both models, especially for FACT, due to its explicit modeling of video transcripts during training. Adding SBL further boosts performance. With MSB and SBL, MSTCN's accuracy improves by 6.5% and F1@50 by 18.4%, while FACT's accuracy improves by 19.9% and F1@50 by 30.6%. Incorporating FBFC further improves the accuracy by 4.9% on MSTCN and the F1@50 by 3.2% on FACT. Finally, integrating all proposed modules leads to the highest performance, resulting in an F1@50 score of 67.4% for MSTCN and 69.9% for FACT. Overall, our approach effectively enhances the capability of standard TAS models to handle multi-task videos.

Training Setup		Acc	Edit	F1@{10,25,50}		
Offline Temporal Action Segmentation						
MSTCN [17]	LMT	78.0	75.9	78.3	76.6	70.4
	LST&LMT	80.6	84.7	84.5	82.5	74.6
Ours	w/o DA	75.7	74.9	79.7	77.6	67.4
	w/ DA	76.6$^{+0.9}$	77.6$^{+2.7}$	82.0$^{+2.3}$	80.6$^{+3.0}$	72.2$^{+4.8}$
	+LMT	82.7	87.3	89.7	88.5	82.4
Online Temporal Action Segmentation						
ProTAS [65]	LMT	61.5	41.1	37.2	32.8	22.0
	LST&LMT	64.5	49.8	39.4	34.5	22.7
Ours	w/o DA	67.8	54.8	54.6	49.8	36.7
	w/ DA	69.1$^{+1.3}$	56.9$^{+2.1}$	57.5$^{+2.9}$	52.0$^{+2.2}$	38.8$^{+2.1}$
	+LMT	74.9	70.6	67.2	62.5	49.3

Table 3. The effect of *domain adaptation* using **unlabeled** multi-task videos and comparison with methods using **labeled** multi-task videos.

Online Temporal Action Segmentation. We evaluate the online TAS performance of our approach in Tab. 2, using ProTAS [65] as the base TAS model. Without access to future frames, the online model relies solely on past and current frames, making it challenging to anticipate action transitions and accurately segment boundaries. This limitation results in a noticeable performance drop compared to the offline models, echoing the difficulties of online MT-TAS. However, our proposed modules still achieve a notable performance boost. When all modules are combined, the model increases accuracy by 18.7% and F1@50 by 24.3% compared to the baseline. Particularly, the integration of FAAR improves the accuracy by 12.6% and all F1 scores by nearly 20%, highlighting the value of foreground-aware refinements in scenarios without access to future frames.

Domain Adaptation with Unlabeled Multi-Task Videos. In Tab. 3, we evaluate our domain adaptation approach using *unlabeled* multi-task videos, as introduced in Sec. 3.8. Domain adaptation consistently improves performance across all metrics, increasing F1@50 by 4.8% and 2.1% in offline and online settings, respectively. To thoroughly assess our method, we compare against two baseline se-

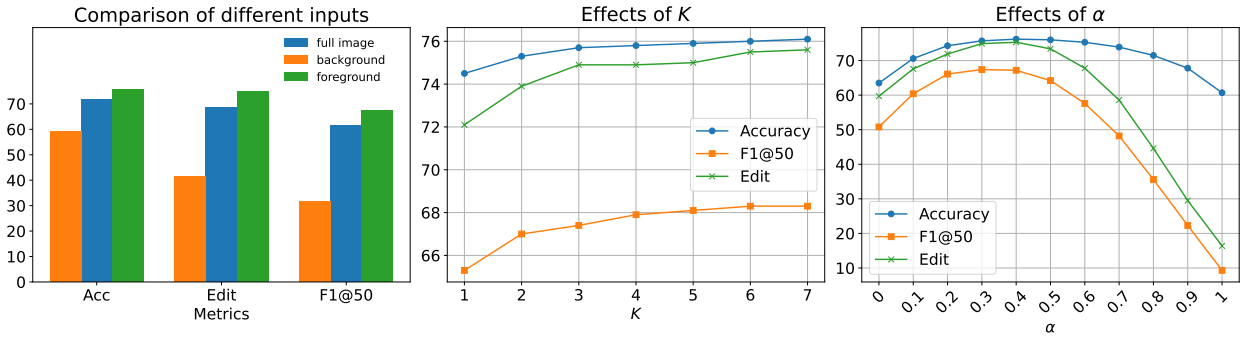


Figure 4. Ablation Studies on FAAR: (a) comparison of different inputs; (b) effects of K; (c) effects of α.

tups using 3-fold cross-validation on multi-task videos: 1) the model is trained on *labeled multi-task (LMT)* videos; 2) the model is trained on both *labeled single-task and labeled multi-task (LST&LMT)* videos. Note that these baselines use labeled multi-task data, while our method employs only unlabeled multi-task videos during training. In offline TAS, our model with domain adaptation outperforms the *LMT* baseline and achieves comparable results to the *LST&LMT* baseline. In the more challenging online setting, our models surpass both baselines, demonstrating the ability of our modules to enhance generalization and handle the data-intensive demands of online TAS. We also extend our method by incorporating labeled multi-task videos (+LMT) to provide an upper bound for domain adaptation. The performance gap reveals room for improvement in unsupervised domain adaptation. Furthermore, the performance gains from *LST&LMT* to *Ours+LMT* demonstrate that our modules provide complementary benefits even when labeled multi-task data is available.

Ablation Studies on FAAR. In Fig. 4, we conduct ablation studies to evaluate different inputs of FAAR and analyze the effects of K and α in Eq. (10) during inference. First, we change the module's input from foreground frames to background frames or full images. Background frames perform poorly while foreground frames outperform full images, showing the importance of action-relevant foreground information. We then examine how varying K, *the number of top predicted action classes considered*, and α, *the weighting factor between initial predictions and foreground-aware predictions*, influences the performance. The model's performance gradually improves with increasing K, reaching a plateau when $K \geq 3$. As α increases from 0 to 1, the model performs best between 0.3 and 0.4. Notably, while the accuracies at $\alpha = 0$ (base TAS predictions only) and $\alpha = 1$ (foreground-aware predictions only) are similar, the Edit and F1@50 scores for foreground-only predictions are significantly lower. This suggests that FAAR emphasizes individual frame accuracy but may lack the temporal consistency needed for segment-wise metrics.

Qualitative Analysis of FBFC. In Fig. 5, we qualitatively

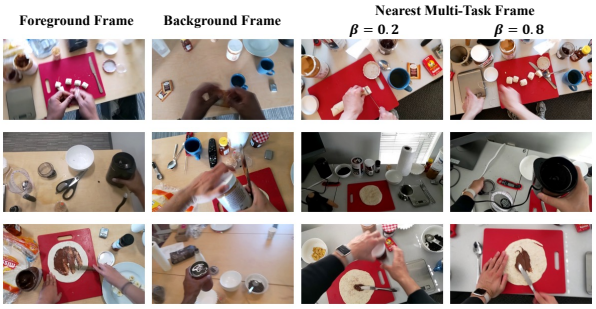


Figure 5. Retrieved nearest multi-task videos by different ratios with FBFC. For simplicity, only middle frames of video clips are displayed.

analyze how FBFC works with different values of the mixing ratio β. We sample two clips from single-task videos, extract foreground frames from one clip and background frames from another, and then recompose these features using FBFC via Eqs. (7) and (8). To assess the recomposition quality, we retrieve the nearest clip from multi-task testing videos based on feature similarity. With a lower mixing ratio ($\beta = 0.2$), the recomposed features are more heavily influenced by background features, and the context of the retrieved frames predominantly match the background frames. Conversely, with a higher mixing ratio ($\beta = 0.8$), the retrieved frames often depict the same action as the foreground frames while sharing some similarities with the context of the background frames. Based on these observations, we sample $\beta \in [0.5, 1]$ in training to ensure preserving action features while introducing background variations.

See supplementary materials for more results, qualitative visualization, and discussion on complexity and limitations.

6. Conclusions

We proposed an MT-TAS framework to tackle the challenges of interleaved actions in multi-task scenarios. Our approach combines synthesized multi-task data, a two-stage TAS model, and the new MEKA benchmark dataset. We validated the effectiveness of our approach through extensive experiments in both offline and online MT-TAS.

Acknowledgement

This work is sponsored by ARPA-H (1AY2AX000062), DARPA PTG (HR00112220001), NSF (IIS-2115110) and ARO (W911NF2110276). Content does not necessarily reflect the position/policy of the Government.

References

[1] Hyemin Ahn and Dongheui Lee. Refining action segmentation with hierarchical video representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16302–16310, 2021. 1, 2

[2] 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

[3] Kyungho Bae, Geo Ahn, Youngrae Kim, and Jinwoo Choi. Devias: Learning disentangled video representations of action and scene for holistic video understanding. In *European Conference on Computer Vision*, 2024. 3

[4] Emad Bahrami, Gianpiero Francesca, and Juergen Gall. How much temporal long-term context is needed for action segmentation? In *International Conference on Computer Vision (ICCV)*, 2023. 2

[5] Siddhant Bansal, Chetan Arora, and C.V. Jawahar. My view is the best view: Procedure learning from egocentric videos. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2

[6] Nadine Behrmann, S. Alireza Golestaneh, Zico Kolter, Juergen Gall, and Mehdi Noroozi. Unified fully and timestamp supervised temporal action segmentation via sequence to sequence translation. In *ECCV*, 2022. 1, 2

[7] 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. 4, 6

[8] 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. 2

[9] Min-Hung Chen, Baopu Li, Yingze Bao, Ghassan AlRegib, and Zsolt Kira. Action segmentation with joint self-supervised temporal domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9454–9463, 2020. 6

[10] Victor G. Turrisi da Costa, Giacomo Zara, Paolo Rota, Thiago Oliveira-Santos, Nicu Sebe, Vittorio Murino, and Elisa Ricci. Dual-head contrastive domain adaptation for video action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1181–1190, 2022. 6

[11] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision (IJCV)*, 130:33–55, 2022. 2

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

[13] Shuangrui Ding, Maomao Li, Tianyu Yang, Rui Qian, Haohang Xu, Qingyi Chen, Jue Wang, and Hongkai Xiong. Motion-aware contrastive video representation learning via foreground-background merging. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9716–9726, 2022. 3

[14] G. Donahue and E. Elhamifar. Learning to predict activity progress by self-supervised video alignment. *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 1

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

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

[17] Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3575–3584, 2019. 1, 2, 4, 6, 7

[18] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2

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

[20] 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

[21] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. *International Conference on Machine Learning*, 2015. 6

[22] Reza Ghoddoosian, Isht Dwivedi, Nakul Agarwal, Chiho Choi, and Behzad Dariush. Weakly-supervised online action segmentation in multi-view instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13780–13790, 2022. 1, 2

[23] Reza Ghoddoosian, Saif Sayed, and Vassilis Athitsos. Hierarchical modeling for task recognition and action segmentation in weakly-labeled instructional videos. *IEEE Winter Conference on Applications of Computer Vision*, 2022.

[24] Reza Ghoddoosian, Isht Dwivedi, Nakul Agarwal, and Behzad Dariush. Weakly-supervised action segmentation and unseen error detection in anomalous instructional videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10128–10138, 2023. 1, 2

[25] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Mar-

tin, Tushar Nagarajan, Ilija Radosavovic, Santhosh K. Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Z. Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina González, James M. Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jáchym Kolár, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran K. Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbeláez, David J. Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard A. Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3,000 hours of egocentric video. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18973–18990, 2021. 2

[26] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19383–19400, 2024. 2

[27] Yifei Huang, Yusuke Sugano, and Yoichi Sato. Improving action segmentation via graph-based temporal reasoning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[28] Yifei Huang, Guo Chen, Jilan Xu, Mingfang Zhang, Lijin Yang, Baoqi Pei, Hongjie Zhang, Lu Dong, Yali Wang, Limin Wang, and Yu Qiao. Egoexolearn: A dataset for bridging asynchronous ego- and exo-centric view of procedural activities in real world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22072–22086, 2024. 2

[29] Yuchi Ishikawa, Seito Kasai, Yoshimitsu Aoki, and Hirokatsu Kataoka. Alleviating over-segmentation errors by detecting action boundaries. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2322–2331, 2021. 2

[30] Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 2

[31] 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. 1, 2

[32] 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. 2

[33] 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. 1, 2

[34] Sateesh Kumar, Sanjay Haresh, Awais Ahmed, Andrey Konin, M. Zeeshan Zia, and Quoc-Huy Tran. Unsupervised action segmentation by joint representation learning and online clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20174–20185, 2022. 2

[35] 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. 2

[36] S. Lee, Z. Lu, Z. Zhang, M. Hoai, and E. Elhamifar. Error detection in egocentric procedural task videos. *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 1, 2, 6

[37] Haoxin Li, Yuan Liu, Hanwang Zhang, and Boyang Li. Mitigating and evaluating static bias of action representations in the background and the foreground. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19911–19923, 2023. 3

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

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

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

[41] M. Li, L. Chen, Y. Duarr, Z. Hu, J. Feng, J. Zhou, and J. Lu. Bridge-prompt: Towards ordinal action understanding in instructional videos. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2

[42] Shi-Jie Li, Yazan AbuFarha, Yun Liu, Ming-Ming Cheng, and Juergen Gall. Ms-tcn++: Multi-stage temporal convolutional network for action segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. 1, 2

[43] Yin Li, Miao Liu, and James M. Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. *European Conference on Computer Vision*, 2018. 2, 6

[44] Zhe Li, Yazan Abu Farha, and Jurgen Gall. Temporal action segmentation from timestamp supervision. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 2

[45] Daochang Liu, Qiyue Li, Anh-Dung Dinh, Tingting Jiang, Mubarak Shah, and Chang Xu. Diffusion action segmentation. *International Conference on Computer Vision*, 2023. 1, 2, 4

[46] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun

Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *eccv*, 2024. 5

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

[48] Z. Lu and E. Elhamifar. Set-supervised action learning in procedural task videos via pairwise order consistency. *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2

[49] Z. Lu and E. Elhamifar. Fact: Frame-action cross-attention temporal modeling for efficient action segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 1, 2, 4, 6, 7

[50] J. Munro and D. Damen. Multi-modal domain adaptation for fine-grained action recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 6

[51] Zhanzhong Pang, Fadime Sener, Shrinivas Ramasubramanian, and Angela Yao. Long-tail temporal action segmentation with group-wise temporal logit adjustment. 2024. 2

[52] Sam Perochon and Laurent Oudre. Unsupervised action segmentation of untrimmed egocentric videos. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 1

[53] Will Price, Carl Vondrick, and Dima Damen. Unweavenet: Unweaving activity stories. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13770–13779, 2022. 2

[54] Camillo Quattrocchi, Antonino Furnari, Daniele Di Mauro, Mario Valerio Giuffrida, and Giovanni Maria Farinella. Synchronization is all you need: Exocentric-to-egocentric transfer for temporal action segmentation with unlabeled synchronized video pairs. *European Conference on Computer Vision*, 2024. 1

[55] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *International Conference on Machine learning*, 2021. 5

[56] Francesco Ragusa, Antonino Furnari, Salvatore Livatino, and Giovanni Maria Farinella. The meccano dataset: Understanding human-object interactions from egocentric videos in an industrial-like domain. *Winter Conference on Applications of Computer Vision*, 2021. 2

[57] Rahul Rahaman, Dipika Singhania, Alexandre Thiery, and Angela Yao. A generalized and robust framework for timestamp supervision in temporal action segmentation. In *Computer Vision–ECCV 2022: 17th European Conference*, 2022. 2

[58] 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.

[59] 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. 2

[60] 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. 2

[61] Saquib Sarfraz, Naila Murray, Vivek Sharma, Ali Diba, Luc Van Gool, and Rainer Stiefelhagen. Temporally-weighted hierarchical clustering for unsupervised action segmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 2

[62] Saif Sayed, Reza Ghoddoosian, Bhaskar Trivedi, and Vassilis Athitsos. A new dataset and approach for timestamp supervised action segmentation using human object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3133–3142, 2023. 2

[63] Fadime Sener, Dibyadip Chatterjee, Daniel Shelepov, Kun He, Dipika Singhania, Robert Wang, and Angela Yao. Assembly101: A large-scale multi-view video dataset for understanding procedural activities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21096–21106, 2022. 2

[64] 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, 2

[65] Y. Shen and E. Elhamifar. Progress-aware online action segmentation for egocentric procedural task videos. *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 1, 2, 6, 7

[66] 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

[67] Y. Shen, H. Wang, X. Yang, M. Feiszli, E. Elhamifar, L. Torresani, and E. Mavroudi. Learning to segment referred objects from narrated egocentric videos. *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 3

[68] 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. 2

[69] 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.

[70] Dipika Singhania, Rahul Rahaman, and Angela Yao. C2f-tcn: A framework for semi-and fully-supervised temporal action segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10):11484–11501, 2023.

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

[72] Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pages 729–738, 2013. 2

[73] Yuhao Su and Ehsan Elhamifar. Two-stage active learning for efficient temporal action segmentation. In *European Conference on Computer Vision*, pages 161–183. Springer, 2024. 1

[74] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[75] Sanket Thakur, Cigdem Beyan, Pietro Morerio, Vittorio Murino, and Alessio Del Bue. Anticipating next active objects for egocentric videos. *IEEE Access*, 2024. 3

[76] Quoc-Huy Tran, Ahmed Mehmood, Muhammad Ahmed, Muhammad Naufil, Anas Zafar, Andrey Konin, and Zeeshan Zia. Permutation-aware activity segmentation via unsupervised frame-to-segment alignment. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6426–6436, 2024. 2

[77] Huiyu Wang, Mitesh Kumar Singh, and Lorenzo Torresani. Ego-only: Egocentric action detection without exocentric transferring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5250–5261, 2023. 1

[78] Jiahao Wang, Guo Chen, Yifei Huang, Limin Wang, and Tong Lu. Memory-and-anticipation transformer for online action understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13824–13835, 2023. 2

[79] Zhe Wang, Hao Chen, Xinyu Li, Chunhui Liu, Yuanjun Xiong, Joseph Tighe, and Charless Fowlkes. Sscap: Self-supervised co-occurrence action parsing for unsupervised temporal action segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1819–1828, 2022. 2

[80] Ming Xu and Stephen Gould. Temporally consistent unbalanced optimal transport for unsupervised action segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 2

[81] Ziwei Xu, Yogesh S Rawat, Yongkang Wong, Mohan Kankanhalli, and Mubarak Shah. Don't pour cereal into coffee: Differentiable temporal logic for temporal action segmentation. In *Advances in Neural Information Processing Systems*, 2022. 6

[82] Lingfeng Yang, Yueze Wang, Xiang Li, Xinlong Wang, and Jian Yang. Fine-grained visual prompting. *Advances in Neural Information Processing Systems*, 36, 2023. 5

[83] 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. 2

[84] Fangqiu Yi, Hongyu Wen, and Tingting Jiang. Asformer: Transformer for action segmentation. In *The British Machine Vision Conference (BMVC)*, 2021. 1, 2, 4, 6

[85] Chuhan Zhang, Ankush Gupta, and Andrew Zisserman. Helping hands: An object-aware ego-centric video recognition model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13901–13912, 2023. 3

[86] Ce Zhang, Changcheng Fu, Shijie Wang, Nakul Agarwal, Kwonjoon Lee, Chiho Choi, and Chen Sun. Object-centric video representation for long-term action anticipation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6751–6761, 2024. 3

[87] Xingyi Zhou, Anurag Arnab, Chen Sun, and Cordelia Schmid. How can objects help action recognition? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2353–2362, 2023. 3

[88] 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. 1, 2