

Online Generic Event Boundary Detection

Hyungrok Jung^{1*} Daneul Kim^{2*§†} Seunggyun Lim¹ Jeany Son^{3†§} Jonghyun Choi^{2†‡}

¹GIST ²Seoul National University ³POSTECH

jhrock2001@gm.gist.ac.kr carpedkm@snu.ac.kr sk000514@gm.gist.ac.kr
jeany@postech.ac.kr jonghyunchoi@snu.ac.kr

Abstract

Generic Event Boundary Detection (GEBD) aims to interpret long-form videos through the lens of human perception. However, current GEBD methods require processing complete video frames to make predictions, unlike humans processing data online and in real-time. To bridge this gap, we introduce a new task, Online Generic Event Boundary Detection (On-GEBD), aiming to detect boundaries of generic events immediately in streaming videos. This task faces unique challenges of identifying subtle, taxonomy-free event changes in real-time, without the access to future frames. To tackle these challenges, we propose a novel On-GEBD framework, ESTimator, inspired by Event Segmentation Theory (EST) [50] which explains how humans segment ongoing activity into events by leveraging the discrepancies between predicted and actual information. Our framework consists of two key components: the Consistent Event Anticipator (CEA), and the Online Boundary Discriminator (OBD). Specifically, the CEA generates a prediction of the future frame reflecting current event dynamics based solely on prior frames. Then, the OBD measures the prediction error and adaptively adjusts the threshold using statistical tests on past errors to capture diverse, subtle event transitions. Experimental results demonstrate that ESTimator outperforms all baselines adapted from recent online video understanding models and achieves performance comparable to prior offline-GEBD methods on the Kinetics-GEBD and TAPOS datasets.

1. Introduction

In the era of diverse video content from platforms such as YouTube, TikTok and Netflix, the growing importance of processing long-form videos has spurred a growth of interest in video understanding tasks. While recent research has

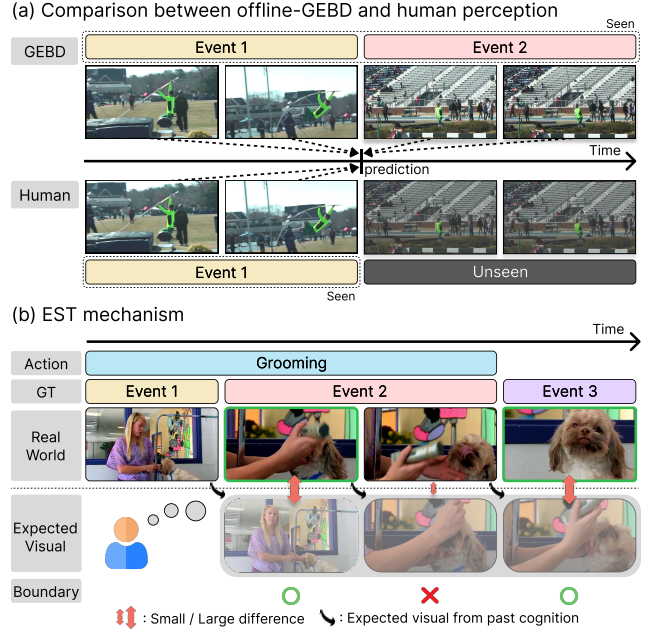


Figure 1. **Comparison of offline-GEBD and human perception with illustration of Event Segmentation Theory (EST).** (a) In a conventional GEBD task, all the event boundaries are determined by utilizing all past and future frames. However, human segments event sequentially relying only on visuals available at the current moment. (b) The illustration of EST shows how humans perceive events. When we perceive visuals, we naturally expect continuous visuals to be recognized. When a significant difference from the given visual input occurs, we perceive it as an event boundary.

actively investigated tasks such as action recognition and action detection [4, 16, 18, 28, 36, 51, 54], these studies typically focus on a limited set of pre-defined action classes and short, trimmed video clips. Consequently, the field of long-form video understanding, which involves analyzing extended video content with complex narratives and diverse actions, remains largely unexplored.

In this regard, Generic Event Boundary Detection (GEBD) [34] introduces a new perspective to understand long-form videos. Originating from cognitive science, the

* Equal Contribution, † Corresponding Authors

§ Work done while at GIST.

‡ Daneul Kim is with CSE in SNU, and was previously with IPAI. Jonghyun Choi is with ECE, ASRI, and IPAI in SNU.

term *event* is an entity of which humans naturally segment continuous visual information, maintaining semantic continuity. As humans perceive broad and diverse visuals, events have the characteristic of being taxonomy-free with various levels of granularity. For example, as illustrated in Figure 1(b), a pre-defined action like *grooming* can be partitioned into multiple sub-events, while frames that do not correspond to any pre-defined action label still be delineated as generic events. By aiming to detect changes between these events, GEBD attempts to analyze long-form videos from a human-like perspective.

However, the current GEBD task aims to identify multiple event boundaries at once, within a fully given chunk of video, which differs significantly from how humans perceive events in an *online* manner (refer to Figure 1(a)). By relying on both past and future frames to determine event boundaries, GEBD does not fully reflect natural human perception, as *humans process visual information from time to time without looking at the future*. To address this limitation and to closely mimic human cognition, we propose a new challenging task, **Online Generic Event Boundary Detection (On-GEBD)**. In On-GEBD, the model must process a streaming video and decide immediately whether each incoming frame is an event boundary, relying solely on past and present information without access to future frames.

On-GEBD not only inherits the challenges from previous offline-GEBD but also faces more complex challenges due to the limited information (*i.e.* past-only information) when determining event boundaries. Prior offline-GEBD methods effectively address the challenges in detecting subtle semantic changes of different events by generating a multi-level difference map among several frames [39] or creating the temporal similarity matrix for the entire video frame [17]. However, due to their reliance on a complete sequence of video frames, these methods are not suitable for On-GEBD, where only past information is available to determine whether a streamed frame is a boundary or not. Also, since an online setting requires immediate determination of boundary as each frame is streamed, it is essential to devise a method specially tailored for On-GEBD.

To address these issues, we propose a simple yet effective framework *ESTimator* inspired by Event Segmentation Theory (EST) in cognitive science. The EST states that humans continuously make predictions consistent with an ongoing event and detect changes when these predictions diverge from the actual information (refer to Figure 1(b)). We design a Consistent Event Anticipator (CEA) module to reflect the essence of the EST. To make CEA predict consistent events robustly, we propose the two training objectives to derive the model to predict visual information consistent with the current ongoing event. *ESTimator* detects event boundaries by assessing the discrepancy between the visual information predicted by CEA and the actual in-

formation. Moreover, determining boundaries based on a fixed threshold presents challenges in distinguishing events that are not constrained by a particular taxonomy and have varying degrees of granularity. Therefore, we propose an Online Boundary Discriminator (OBD) module that determines boundaries by comparing the distribution of visual discrepancies from the surrounding frames. OBD stores historical discrepancies in a fixed-size queue and conducts statistical testing to provide a dynamic threshold that reflects the surrounding context.

Our method successfully tackles the unique challenges of On-GEBD and shows its effectiveness on two GEBD benchmark datasets, Kinetics-GEBD [34] and TAPOS [32]. We show that our model not only outperforms baseline methods that utilize existing online video understanding methods [2, 41, 44, 54], but also achieves comparable results to prior offline methods that are tested under the original GEBD setting.

We summarize our contribution as follows:

- We present a new challenging task, On-GEBD, designed to align closer to the actual human perception.
- To address the unique challenges posed by On-GEBD, we propose a novel framework, *ESTimator*, comprising with a Consistent Event Anticipator (CEA) and Online Boundary Discriminator (OBD).
- Our model outperforms various baselines based on traditional online video models and achieves performance on par with offline setting [34, 45].

2. Related Work

Generic Event Boundary Detection. Generic Event Boundary Detection (GEBD), introduced in [34], aims to detect event boundaries aligning with human perception. Unlike traditional video understanding tasks (*e.g.*, TAL [25, 51, 53], Action Recognition [15, 42]), GEBD deals with continuous semantics without taxonomy, enabling the understanding of complex video. Recent GEBD works aim to detect boundaries by processing entire videos or analyzing surrounding frame context. For instance, UBoCo [17] uses temporal similarity matrices and contrastive learning, while DDM-Net [39] employs progressive attention on multi-level dense difference maps. CoSeg [45], inspired by cognitive modeling [50], introduces boundary detection through event reconstruction. While offline-GEBD methods [12, 17, 23, 24, 31, 38, 39, 45, 52, 55, 56] have shown improvements, our work adapts GEBD to an online setting, addressing the challenge of making instant decisions without future frames.

Video Understanding. Video understanding encompasses various tasks, including Temporal Action Detection (TAD) [27, 33, 47], Temporal Action Localization (TAL) [25, 51, 53], and Video Instance Segmenta-

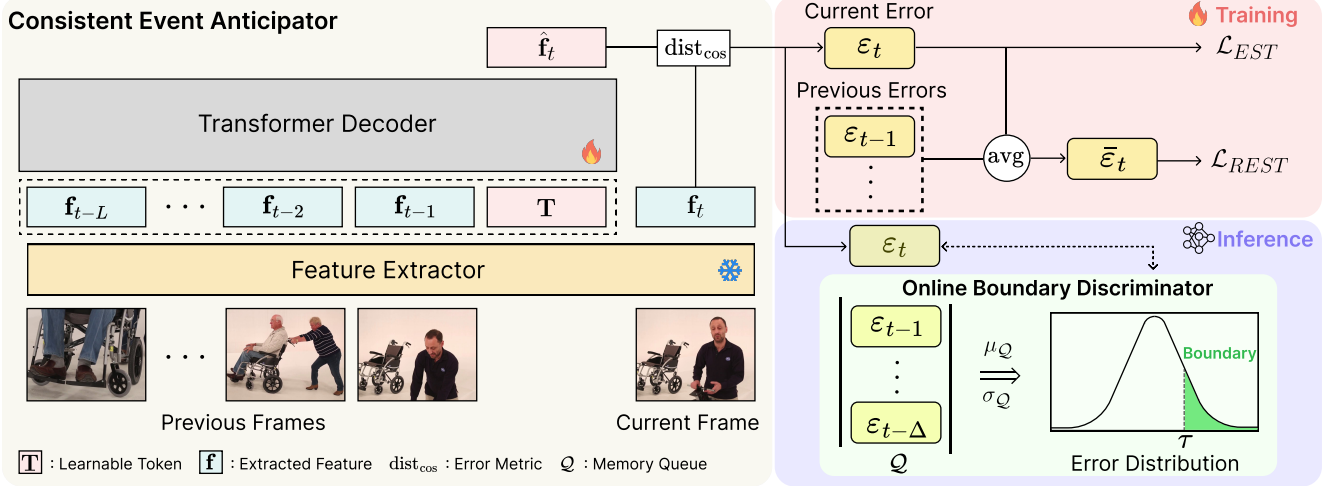


Figure 2. **Overview of our *ESTimator* framework.** Our framework consists of three major components: Consistent Event Anticipator (CEA) which generates a consistent future frame feature using a learnable token (Left). EST-inspired training objective that accumulates frame-level (EST loss) and region-level (REST loss) prediction errors derived from the discrepancy between the generated future frame from CEA with the actual input frame (Upper right). Online Boundary Discriminator (OBD) with a queue that stores past error prediction values to conduct statistical testing on the error derived from the current input frame for inference (Lower right).

tion (VIS) [20, 49]. Recent advancements utilize transformers [51], memory structures [5], and end-to-end approaches [25]. However, these tasks often rely on predefined classes, limiting real-world applicability. While open-set tasks in TAL [3] have been proposed, they still focus on class-based localization. GEBD [34] addresses this limitation by dealing with taxonomy-free events.

Online Video Understanding. Online video understanding focuses on streaming content, sharing commonalities with On-GBD. Tasks include Online Action Detection (OAD) [7], Online Temporal Action Localization (On-TAL) [16], and Online Detection of Action Start (ODAS) [35]. Recent works like LSTR [48], TeSTra [54], and MiniROAD [2] have demonstrated the effectiveness of transformers and GRUs in OAD. For On-TAL, methods like CAG-QIL [16] and OAT [21] have been proposed. While ODAS may seem similar to On-GBD, it differs significantly as it deals with fixed action classes. On-GBD focuses on detecting subtle changes among taxonomy-free events, requiring a distinct approach from conventional online video understanding methods.

3. Online Generic Event Boundary Detection

In this section, we present a new task of Online Generic Event Boundary Detection (On-GBD), focusing on the identification of event boundaries in streaming videos, which is distinct from the traditional offline-GBD task.

The offline-GBD [34] considers a video consisting of N frames, $V = \{v_t\}_{t=1}^N$, segmented into multiple events with M distinct generic event boundaries denoted as $\mathcal{B} =$

$\{b_j\}_{j=1}^M$, where b_j represents the timestamp of the j -th event boundary. However, unlike the human perception process, where incoming visual information is segmented into events instantaneously, the offline-GBD task allows to utilize all frames in V to determine boundaries b_j .

The On-GBD task imposes two key constraints on the conventional offline-GBD: (1) the video is streamed sequentially frame-by-frame; (2) the model must make immediate decisions for each incoming frame on whether v_t is an event boundary as soon as the frame is received. These constraints limit the model to use only the past and current frames (*i.e.*, $v_1 \sim v_t$) for decision-making, without access to the future context of $v_{t+1} \sim v_N$. This online, constrained setting makes the model to closely resemble the real-time, causal nature of human event perception.

By virtue of its online setting, On-GBD presents a novel and more demanding set of challenges than the previous offline-GBD task. The absence of future context exacerbates the issues inherited from offline-GBD in detecting diverse, generic event boundaries. Since the model is compelled to make instant decisions based on limited information, it must balance rapid boundary detection with the risk of false positives. Consequently, On-GBD necessitates the development of novel algorithms that robustly detect the event transitions in streaming visual data with low-latency, which closely aligns with human perception.

4. Method

Our framework, *ESTimator*, addresses the unique challenges of On-GBD by drawing inspiration from the Event

Segmentation Theory (EST), which explains human event perception [30]. ESTimator incorporates two key principles from the EST: (1) humans continuously anticipate consistent information about ongoing events; (2) perceive event changes when there is a significant discrepancy between anticipated and actual information (refer to Figure 1(b)). These principles offer an effective approach to overcome the key challenges of On-GEBD. The continuous anticipation of event information allows the model to make decisions, even without access to future frames. Moreover, by focusing on the trends in discrepancies between anticipated and actual information, the model can effectively detect subtle and diverse event boundaries without relying on pre-defined taxonomies, thereby addressing the ambiguous and varied nature of generic events in streaming video.

Regarding the principles, our framework consists of two components: Consistent Event Anticipator (CEA) and the Online Boundary Discriminator (OBD). CEA is trained with two novel training objectives to anticipate consistent visual information of the ongoing event robustly. Since the consistent information anticipated from CEA diverges from actual information at event boundaries, we leverage this discrepancy as a reliable cue for event boundary detection in online scenario. For clarity, we define the term **error** to denote the discrepancy between the actual and anticipated visual information in the following section. Also, OBD incorporates prior errors to effectively detect diverse forms of event boundaries, enabling more refined and precise detection based on prior semantic changes. In the following section, we explore each component in detail.

4.1. Consistent Event Anticipator

Transformers have proven their effectiveness in handling sequential data for video understanding tasks, particularly in predicting future states based on past and present information [9, 10, 37, 40, 46]. Especially, transformer decoders endowed with auto-regressive causal masking excel at next-information prediction, as exemplified by Large Language Models (LLMs). In this context, we utilize the transformer decoder [43] in Consistent Event Anticipator (CEA) to construct a model that anticipates consistent frame information aligned with ongoing events.

Transformer Decoder. Following a previous GEBD work [17], we first extract frame features, $\mathbf{f}_t \in \mathbb{R}^D$, using a pre-trained ResNet-50 image encoder [13]. We concatenate a single learnable token $\mathbf{T} \in \mathbb{R}^D$ with L extracted frame features $\mathbf{F}_t = \{\mathbf{f}_i\}_{i=t-L}^{t-1}$ and forward them into transformer decoder layers $\mathcal{M}_\theta$ to predict streamed frame features.

$$\begin{aligned} \mathbf{X}_t &= \text{concat}(\mathbf{F}_t, \mathbf{T}), \\ \hat{\mathbf{f}}_t &= \mathcal{M}_\theta(\mathbf{X}_t), \end{aligned} \quad (1)$$

In Eq. 1, $\hat{\mathbf{f}}_t$ denotes the output of the learnable token $\mathbf{T}$ after processing through $\mathcal{M}_\theta$, which encapsulates a prediction for the upcoming frame feature (refer Figure 2). With the causal-attention mask in the transformer decoder, we ensure that succeeding tokens only attend to the preceding ones.

Objective Functions. The main objective of CEA is to maximize errors at semantically inconsistent event boundaries while minimizing errors within consistent event segments, thereby embodying the core principles of EST. We preliminarily define the error ε_t between the prediction $\hat{\mathbf{f}}_t$ and the actual frame feature $\mathbf{f}_t$ with the cosine distance, and scale it between 0 and 1 as follows:

$$\varepsilon_t = \text{dist}_{\cos}(\mathbf{f}_t, \hat{\mathbf{f}}_t) = \frac{1}{2} \left(1 - \frac{\mathbf{f}_t \cdot \hat{\mathbf{f}}_t}{\|\mathbf{f}_t\| \|\hat{\mathbf{f}}_t\|} \right). \quad (2)$$

To achieve the objective of CEA following EST, we use a binary cross-entropy loss for our **EST loss**, as follows:

$$\mathcal{L}_{EST}(\varepsilon_t, t) = -y_t \log \varepsilon_t - (1 - y_t) \log(1 - \varepsilon_t), \quad (3)$$

where $y_t = 1$ if $t \in \mathcal{B}$, otherwise 0. The EST loss encourages $\mathcal{M}_\theta$ to maximize the errors at event boundaries while minimizing them elsewhere, effectively distinguishing boundary frames from non-boundary frames.

However, strict frame-wise binary supervision would be sub-optimal in videos where consecutive frames contain continuous semantic flow. This is because consecutive frames are smoothly connected without abrupt changes, except shot changes. To address this issue, we propose a region-level training scheme that considers temporal context flow. Since this training scheme shares the mechanism of the EST loss, we named it **REST loss (Region EST loss)**. Incorporating errors derived from nearby regions, REST loss aims to give soft supervision of the abrupt label transitions in the near future. Assuming the size of a region as K , we collect the series of errors $\varepsilon_{t-K}, \dots, \varepsilon_t$ from the inputs $\mathbf{X}_{t-K}, \dots, \mathbf{X}_t$, and compute the average of these consecutive past errors $\bar{\varepsilon}_t$ as follows:

$$\bar{\varepsilon}_t = \frac{1}{K} \sum_{i=t-K}^t \varepsilon_i. \quad (4)$$

The REST loss is then defined as follows:

$$\mathcal{L}_{REST}(\varepsilon_t, t) = \mathcal{L}_{EST}(\bar{\varepsilon}_t, t), \quad (5)$$

where the boundary label at t is used for the loss. The REST loss allows the predicted frame features, $\hat{\mathbf{f}}_{t-K} \dots \hat{\mathbf{f}}_t$, to be softly trained with additional future information, making them less sensitive to noise and more effective in anticipating future events.

Our final loss function is computed as a weighted sum of two losses, EST and REST losses, as follows:

$$\mathcal{L}(\varepsilon_t, t) = \alpha \cdot \mathcal{L}_{REST}(\varepsilon_t, t) + \sum_{i=t-K}^t \mathcal{L}_{EST}(\varepsilon_i, i). \quad (6)$$

In our experiments, we set the hyperparameter α to 0.5.

Batch-wise Loss Weighting. In the video, frames that correspond to boundaries account for a significantly smaller proportion compared to non-boundary frames. Previous offline-GEBD studies utilize balanced samplers [34, 39] or weighted loss terms [14] with manually tuned values to alleviate issues arising from such imbalanced data distribution. To achieve a similar effect during training, we utilize batch-wise loss weighting technique. We first calculate the ratio of boundary and non-boundary targets within a single batch. This ratio is then multiplied by the loss calculated for boundary targets, allowing the batch-wise loss weighting to balance the impact of boundary and non-boundary samples during training. This approach not only eliminates the need for manually tuned scaling values but also aims to achieve a similar effect to that of using a batch sampler.

4.2. Online Boundary Discriminator

Conventional offline-GEBD methods typically rely on either static thresholds [14, 34, 39] or dynamic criteria based on peak detection [45] to identify boundaries by analyzing the entire sequence of frames—including future ones. However, these approaches are not well-suited to the On-GEBD: static thresholds fail to capture diverse form of semantic changes, and peak detection is impractical due to its reliance on future frame information. These limitations highlight the need for a novel approach tailored to the immediacy and dynamic nature of On-GEBD.

To address these unique challenges, we propose an Online Boundary Discriminator (OBD), which dynamically discriminates event boundaries by leveraging recent semantic flow. First, OBD stores errors up to the current frame in a fixed-size memory queue, denoted as $\mathcal{Q}$, which is managed in a First-In, First-Out (FIFO) manner. When the t -th frame v_t is streamed, $\mathcal{Q}$ contains past errors $\varepsilon_{t-\Delta}, \dots, \varepsilon_{t-1}$, where Δ is the size of the $\mathcal{Q}$. Using the errors stored in a memory queue, OBD conducts statistical testing with the Gaussian distribution derived from errors in the $\mathcal{Q}$. By normalizing the incoming error ε_t with statistical information derived from $\mathcal{Q}$, the OBD determines a boundary if the incoming error is identified as an outlier compared to the close region errors. Figure 3 exemplifies how the underlying mechanism of OBD dynamically modulates the detection threshold.

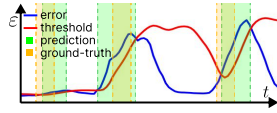


Figure 3. An illustration of how Online Boundary Detector (OBD) applies a dynamic threshold to capture diverse event transitions.

$$\zeta_t = \frac{\varepsilon_t - \mu_{\mathcal{Q}}}{\sigma_{\mathcal{Q}}}, \quad (7)$$

$$\text{OBD}(\mathcal{Q}, \varepsilon_t) = \mathbb{1}[\zeta_t > \tau],$$

In Eq. 7, τ is a threshold value for an indicator function $\mathbb{1}[\cdot]$ and ζ_t is the normalized error at frame t by the mean $\mu_{\mathcal{Q}}$

and the standard deviation $\sigma_{\mathcal{Q}}$ derived from errors stored in $\mathcal{Q}$ (refer lower right side of the Figure 2).

In our experiments, we set τ to 1.5 empirically. By dynamically adjusting the error threshold through OBD, our ESTimator successfully identifies a wide range of event boundaries, as demonstrated in Figure 4.

5. Experiment

5.1. Setup

Benchmark Dataset. The Kinetics-GEBD dataset [34] is composed of approximately 60K videos selected from the Kinetics-400 dataset [19]. The selected videos are divided into units of taxonomy-free events by specially trained annotators. On average, each video contains about 5 different events. The train, validation and test videos are almost equally distributed with 18,794, 18,813 and 17,725 videos, respectively. Since annotations for the test set are not available, we report the results evaluated on the validation set, following prior works [17, 34, 39]. For cross-validation, we randomly partitioned the dataset into training (80%) and validation (20%) subsets for the experiments. TAPOS dataset [32] is composed of Olympic sports videos that are annotated with 21 different actions with 13,094 training action instances and 1,790 validation action instances. Following [34], we re-purpose TAPOS for the GEBD task by obscuring the action labels of sub-actions. Additionally, we present results for INRIA [1] in supplementary material.

Evaluation Metric. Relative Distance (Rel.Dis) is a metric that measures the relative difference between the detected boundary timestamps and the ground truth boundary timestamps in GEBD. We calculate the metric by dividing the distance between predictions and ground-truths by the union of predicted and ground-truth event instances. When the model predicts consecutive frames as boundaries, as discussed in [34], we set the center of these frames as the predicted boundary timestamps. Relative distance (Rel.Dis) of detected timestamps is evaluated under 10 thresholds with intervals of 0.05, ranging from 0.05 to 0.5. We consider the result to be positive if it is below each threshold. Following the prior works, we report the F1 scores on each threshold, as well as their average in our main experiments.

Baseline. Directly extending the offline-GEBD methods into an online setting poses challenges as they are not designed to process streaming videos. Therefore, we utilize state-of-the-art online video understanding models, which are natively designed for action detection or localization, as our baseline for On-GEBD. Specifically, we utilize TeS-Tra [54], OadTR [44], and MiniROAD [2] from Online Action Detection, and Sim-On [41] from Online Temporal Action Localization as our naïve baselines. Only the head of each model is modified to perform binary classification

Table 1. **Quantitative comparison with other online baselines in On-GEBD task.** BC denotes a binary classifier attached to the last layer of each model to solve On-GEBD. We denoted with **bold** for the highest F1 score and second with underline.

Dataset	Rel. Dis. threshold	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	Avg
Kinetics-GEBD	TeSTra - BC [54]	0.438	0.488	0.521	0.545	0.564	0.580	0.593	0.604	0.614	0.622	0.557
	Sim-On - BC [41]	0.461	0.534	0.579	0.610	0.633	0.651	0.664	0.675	0.685	0.692	0.618
	OadTR - BC [44]	0.474	0.512	0.535	0.552	0.565	0.575	0.583	0.590	0.596	0.601	0.558
	MiniROAD - BC [2]	<u>0.569</u>	<u>0.622</u>	<u>0.649</u>	<u>0.675</u>	<u>0.691</u>	<u>0.704</u>	<u>0.714</u>	<u>0.722</u>	<u>0.729</u>	<u>0.735</u>	<u>0.681</u>
	ESTimator (Ours)	0.620	0.687	0.724	0.746	0.762	0.774	0.782	0.789	0.795	0.799	0.748
TAPOS	TeSTra - BC [54]	0.364	0.417	0.452	0.478	0.496	0.511	0.523	0.533	0.542	0.550	0.487
	Sim-On - BC [41]	0.225	0.269	0.303	0.329	0.350	0.367	0.381	0.394	0.405	0.415	0.344
	Oad-TR - BC [44]	0.263	0.319	0.361	0.394	0.422	0.445	0.465	0.483	0.497	0.510	0.416
	MiniROAD - BC [2]	0.422	0.472	0.502	<u>0.522</u>	<u>0.537</u>	<u>0.549</u>	<u>0.558</u>	<u>0.566</u>	<u>0.572</u>	<u>0.578</u>	<u>0.528</u>
	ESTimator (Ours)	<u>0.394</u>	<u>0.455</u>	<u>0.499</u>	0.532	0.558	0.578	0.594	0.608	0.619	0.629	0.547

Table 2. **Quantitative comparison with offline methods.** Note that we report the performance of the models in offline setting from their original literature. Also, we indicate the highest F1 score with **bold**, second with underline and third with †.

Dataset	Setting	Supervision	Rel. Dis. threshold	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	Avg
Kinetics-GEBD	Offline	Supervised	BMN [26]	0.186	0.204	0.213	0.220	0.226	0.230	0.233	0.237	0.239	0.241	0.223
			BMN-StartEnd [34]	0.491	0.589	0.627	0.648	0.660	0.668	0.674	0.678	0.681	0.683	0.640
			TCN-TAPOS [34]	0.464	0.560	0.602	0.628	0.645	0.659	0.669	0.676	0.682	0.687	0.627
			TCN [22]	0.588	0.657	0.679	0.691	0.698	0.703	0.706	0.708	0.710	0.712	0.685
			PC [34]	0.625	0.758	0.804	0.829	0.844	0.853	0.859	0.864	0.867	0.870	0.817
	Offline	Unsupervised	SceneDetect [34]	0.275	0.300	0.312	0.319	0.324	0.327	0.330	0.332	0.334	0.335	0.318
			PA-Random [34]	0.336	0.435	0.484	0.512	0.529	0.541	0.548	0.554	0.558	0.561	0.506
			PA [34]	0.396	0.488	0.520	0.534	0.544	0.550	0.555	0.558	0.561	0.564	0.527
			CoSeg [45]	0.656	0.758	<u>0.783</u>	<u>0.794</u>	<u>0.799</u>	<u>0.803</u>	<u>0.804</u>	<u>0.806</u>	<u>0.807</u>	<u>0.809</u>	<u>0.782</u>
	Online	Supervised	ESTimator (Ours)	0.620 [†]	0.687 [†]	0.724 [†]	0.746 [†]	0.762 [†]	0.774 [†]	0.782 [†]	0.789 [†]	0.795 [†]	0.799 [†]	0.748 [†]
TAPOS	Offline	Supervised	ISBA [34]	0.106	0.170	0.227	0.265	0.298	0.326	0.348	0.348	0.348	0.348	0.330
			TCN [34]	0.237	0.312	0.331	0.339	0.342	0.344	0.347	0.348	0.348	0.348	0.330
			CTM [34]	0.244	0.312	0.336	0.351	0.361	0.369	0.374	0.381	0.383	0.385	0.350
			TransParser [22]	0.289	0.381	0.435	0.475	0.500	0.514	0.527	0.534	0.540	0.545	0.474
			PC [34]	0.522	0.595	0.628	0.646	0.659	0.665	0.671	0.676	0.679	0.683	0.642
	Offline	Unsupervised	SceneDetect [34]	0.035	0.045	0.047	0.051	0.053	0.054	0.055	0.056	0.057	0.058	0.051
			PA-Random [34]	0.158	0.233	0.273	0.310	0.331	0.347	0.357	0.369	0.376	0.384	0.314
			PA [34]	<u>0.360</u>	<u>0.459</u>	<u>0.507</u>	<u>0.543</u>	<u>0.567</u>	<u>0.579</u>	0.592 [†]	0.601 [†]	0.609 [†]	0.615 [†]	0.543 [†]
	Online	Supervised	ESTimator (Ours)	0.394 [†]	0.455 [†]	0.499 [†]	0.532 [†]	0.558 [†]	0.578 [†]	<u>0.594</u>	<u>0.608</u>	<u>0.619</u>	<u>0.629</u>	<u>0.547</u>

(BC), as it is necessary for the model to distinguish whether a streamed input is a boundary or not.

Implementation Details. We preprocess the video data by sampling at 24 FPS for the Kinetics-GEBD dataset and 6 FPS for the TAPOS dataset. Following [17], we use the features extracted from the ResNet-50 encoder, which is pre-trained on ImageNet [8] with feature dimension $D = 2,048$. For our experiments, we employ 3 transformer decoder layers in both datasets. Additionally, we train our model with a batch size of 512 using the AdamW [29] optimizer in training with a learning rate of $1e-4$.

5.2. Main Result

Table 1 shows that our framework outperforms baseline models for On-GEBD on both Kinetics-GEBD and TAPOS datasets. Traditional online methods like TeSTra-BC, Oad-

TR-BC, and Sim-On-BC show a lack of model capacity in learning to discriminate generic event boundaries, limited by their model design to learn pre-defined action classes. While MiniROAD-BC demonstrates higher performance than other baselines, our approach still surpasses them on both datasets. The results highlight that simply adapting existing approaches is insufficient; a dedicated method like ESTimator is necessary to detect subtle semantic changes during the event transition.

In Table 2, we compare our framework with models evaluated in an offline setting. ESTimator performs on par with or exceeds most offline methods, achieving higher Avg. F1 scores on Kinetics-GEBD—except for PC [34] and CoSeg [45]. Similarly, on the TAPOS, ESTimator outperforms all baselines from the original GEBD, with the sole exception of the PC method [34].

Table 3. **Benefit of the proposed components.** As a baseline, we utilize the transformer decoder with binary classifier. We gradually add each of the proposed components to investigate its effect on improving performance.

Method	① EST	② REST	③ OBD	F1 @ 0.05	Avg F1
Baseline	✗	✗	✗	0.483	0.607
①	✓	✗	✗	0.571	0.698
②	✗	✓	✗	0.504	0.654
①+②	✓	✓	✗	0.544	0.691
①+③	✓	✗	✓	0.604	0.659
②+③	✗	✓	✓	0.621	0.692
①+②+③ (Ours)	✓	✓	✓	0.620	0.748

Table 4. **Comparison on different metric for calculating the error.** We compare Avg. F1 scores using different distance metrics, with min-max normalization applied to each batch during training. Cosine distance is the best choice due to its bounded nature.

Error Calculation Metric	Avg F1
L1 Distance (min/max normalized)	0.733
L2 Distance (min/max normalized)	0.733
KL Divergence (min/max normalized)	0.734
Cosine Distance (Ours)	0.748

5.3. Ablation Study

Impact of Proposed Components. To observe the benefits of each proposed component, we build our components on top of the simple transformer model with a binary classifier, which we refer to as *Baseline* in Table 3. The experimental results present the effectiveness of error-based detection of event boundaries, as the model trained with either EST or REST loss consistently outperforms the baseline.

Naïvely applying the REST loss on top of the EST loss degrades the models’ performance, as using both objectives together tends to reduce errors on boundary frames. Furthermore, the OBD proved to be less effective when applied to a CEA trained exclusively with either EST or REST loss. The performance gap between our full proposed framework and its ablated versions demonstrates the synergistic effect of our individual components.

Metric for Error Calculation. We investigate the impact of the metric used for error computation on the performance of our framework. Unlike the cosine distance, which is bounded, other widespread metrics such as the L1/L2 distance and the KL Divergence are generally unbounded. Therefore, to facilitate the application of our proposed EST and REST losses, we experiment these metrics with min-max normalization per batch while training the CEA. As shown in Table 4, the cosine distance demonstrates superior performance compared to the L1/L2 distance and the KL divergence. Furthermore, we note that these alternative metrics remain relatively viable due to our proposed OBD. OBD provides reliable criteria even for error values with

Table 5. **Comparison of real-time performance (in FPS) with other baselines, which utilize other online video understanding models.** We denoted with bold for the highest FPS and performance, and underlined for the second highest. ESTimator shows the highest Avg. F1 with compatible overall FPS compared to MiniROAD-BC [2]. All experiments were conducted on a single NVIDIA RTX A6000 GPU.

Method	RGB Feat	Model	Overall	Avg F1
TeSTra - BC		177	72.5	0.557
Sim-On - BC		275	76.3	0.618
OadTR - BC	181	100	48.9	0.558
MiniROAD - BC		3069	99.8	<u>0.681</u>
ESTimator (Ours)		<u>481</u>	<u>96.3</u>	0.748

Table 6. **Video feature comparison.** We compare the performance of our method and baselines using video features extracted from TSN networks pre-trained on Something Something v2 (SS-v2) and Kinetics datasets.

Method	Backbone	Avg F1	Backbone	Avg F1
TeSTra - BC		0.653		0.654
Sim-On - BC		0.524		0.501
OadTR - BC	TSN	0.700	TSN	0.699
MiniROAD - BC	(SS-v2)	0.684	(Kinetics)	0.689
ESTimator (Ours)		0.741		0.744

Table 7. **Comparison on outlier handling in OBD.** Removing outliers from the queue significantly reduces performance, demonstrating our OBD functions as a dynamic threshold adaptation.

OBD Setting	Avg F1
Using Only Inliers	0.663 (-11.4%)
Full OBD (Ours)	0.748

unbounded ranges, thereby mitigating the challenges of selecting an appropriate threshold.

Real Time Performance. We report a runtime analysis of our method with other baselines in Table 5 and demonstrate the feasibility of our approach for real-time processing. For all methods, we utilize the ResNet-50 encoder pre-trained on ImageNet [8], which operates at 181 FPS in our experimental setting. ESTimator not only demonstrates superior performance, but also achieves higher FPS compared to other transformer-based baselines (*i.e.*, TeSTra-BC, Sim-On-BC, OadTR-BC). Even with superior performance, its overall FPS is on par with that of MiniROAD-BC, which is based on GRU [6] architecture. We further demonstrate the analysis on computation cost in supplementary material.

Utilizing Video Feature. To ensure a fair comparison, we utilize ResNet-50 as the feature extractor, in accordance with previous offline-GBED studies. However, since the baseline models were originally developed with different feature extractors and may underperform with ResNet-50

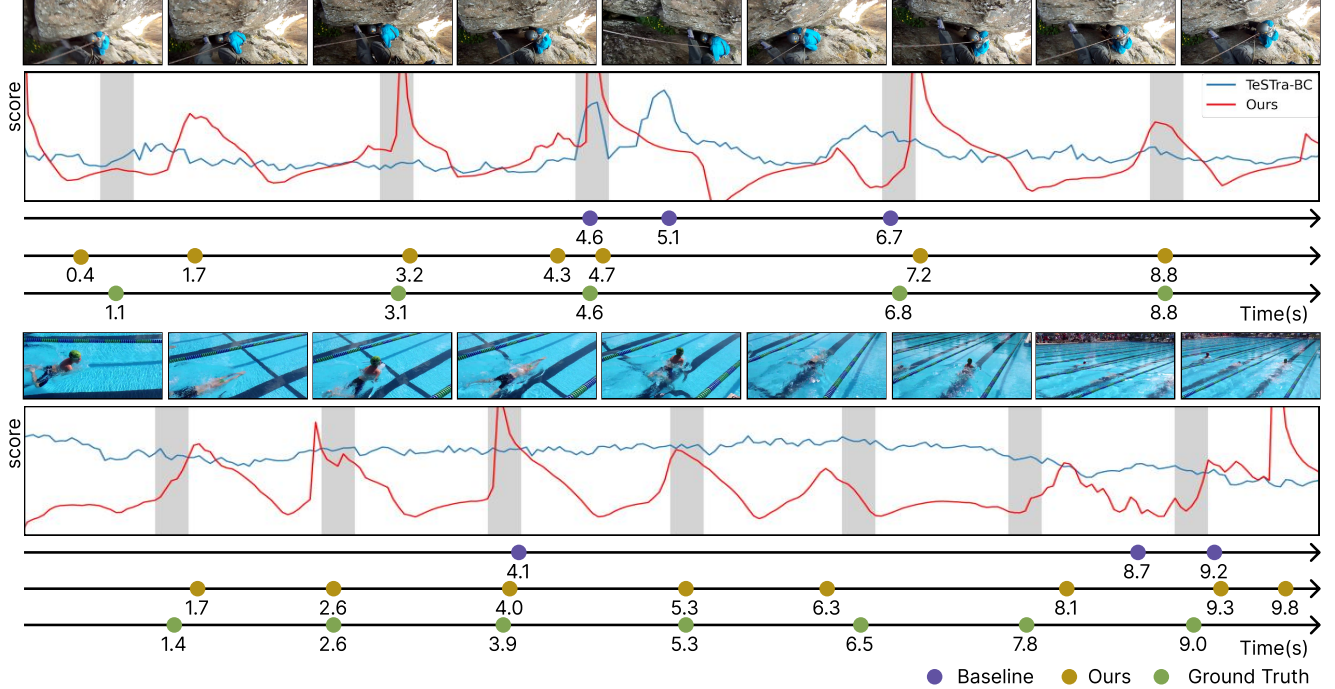


Figure 4. **Qualitative result.** Comparison between our proposed framework and baseline from other online video understanding task. Note that baseline here refers to TeSTra [54] with binary classifier head attached for event boundary detection.

features. Table 6 presents the results using the TSN network as a backbone, which was originally used by these baselines. Despite utilizing different backbone architectures, our framework still outperforms all other baselines, thereby proving its robust effectiveness.

Error Handling in OBD. A key challenge in On-GEBD is balancing sensitivity to upcoming boundaries while preventing false alarms. OBD addresses this challenge by storing past errors to adjust its sensitivity to boundaries. However, the persistence of elevated error values from a previous event boundary within the queue may lead to interference, hampering the detection of subsequent event boundaries. To investigate this potential issue, we conducted an ablation study in which error values identified as event boundaries were excluded from the OBD queue. As shown in Table 7, excluding outliers severely degraded performance, reducing the Avg F1 from 0.748 to 0.663 (11.4% relative drop). This finding denotes that including errors from event boundaries is an essential mechanism for robust event boundary detection in OBD while ensuring computational efficiency. We believe that one factor contributing to the effectiveness of this mechanism is its cognitive plausibility, as human perception adopts adaptive criteria for event segmentation when exposed to rapid changes [11].

5.4. Qualitative Result

In Figure 4, we present qualitative comparisons with the TeSTra-BC baseline. In contrast to the baseline, which

shows noisy predictions as in all of the examples in Figure 4, ESTimator detects boundaries more accurately that align closely with the ground truth. Specifically, in the first example (upper panel), there are 5 ground-truth boundaries; ESTimator identifies spikes at 4 of these boundaries within the gray-shaded ground-truth area, while TeSTra-BC detects only 3 boundaries—one of which falls inside an event segment rather than on an actual boundary. In the second example, which contains 7 ground-truth regions, TeSTra-BC successfully detects only 2 boundaries, whereas our method predicts 8 boundaries, 7 of which are correct.

6. Conclusion

We introduce a challenging new task, On-GEBD, designed to bring GEBD closer to human perceptual processes. To address this task, we present ESTimator, inspired by Event Segmentation Theory (EST) from cognitive science, which explains how humans perceive and segment events. Our Consistent Event Anticipator (CEA) is trained using two losses—EST loss and REST loss—to effectively predict future frames consistent with current events, thereby maximizing errors at event boundaries. Additionally, the On-line Boundary Discriminator (OBD) employs dynamic criteria to distinguish boundaries based on error values produced by the CEA. Our work demonstrates superior performance compared to baselines adapted from other online video understanding tasks and shows almost comparable performance to some recent work in the offline-GEGB task.

Acknowledgments. This work was supported by the IITP grants (RS-2019-II191842, RS-2021-II212068, RS2022-II220926 (30%), RS-2022-II220077, RS-2022-II220113, RS-2022-II220959, RS-2022-II220871, RS-2025-02263598 (10%), RS-2021-II211343 (SNU AI), RS-2021-II212068 (AI Innovation Hub), RS-2025-25442338 (AI Star Fellowship-SNU)) funded by MSIT, and the GIST-MIT Research Collaboration grant funded by GIST (10%), Korea.

References

- [1] Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. Learning from narrated instruction videos. In *CVPR*, 2016. 5, 3
- [2] Joungbin An, Hyolim Kang, Su Ho Han, Ming-Hsuan Yang, and Seon Joo Kim. Miniroad: Minimal rnn framework for online action detection. In *ICCV*, pages 10341–10350, 2023. 2, 3, 5, 6, 7
- [3] Wentao Bao, Qi Yu, and Yu Kong. Opental: Towards open set temporal action localization. In *CVPR*, pages 2979–2989, 2022. 3
- [4] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, page 4, 2021. 1
- [5] Feng Cheng and Gedas Bertasius. Tallformer: Temporal action localization with a long-memory transformer. In *ECCV*, pages 503–521. Springer, 2022. 3
- [6] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014. 7
- [7] Roeland De Geest, Efstratios Gavves, Amir Ghodrati, Zhenyang Li, Cees Snoek, and Tinne Tuytelaars. Online action detection. In *ECCV*, pages 269–284, 2016. 3
- [8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 6, 7
- [9] Rohit Girdhar and Kristen Grauman. Anticipative video transformer. In *ICCV*, pages 13505–13515, 2021. 4
- [10] Dayoung Gong, Joonseok Lee, Manjin Kim, Seong Jong Ha, and Minsu Cho. Future transformer for long-term action anticipation. In *CVPR*, pages 3052–3061, 2022. 4
- [11] Urit Gordon, Shimon Marom, and Naama Brenner. Visual detection of time-varying signals: Opposing biases and their timescales. *PLOS ONE*, 14(11):e0224256, 2019. 8, 2
- [12] Sourabh Vasant Gothe, Vibhav Agarwal, Sourav Ghosh, Jayesh Rajkumar Vachhani, Pranay Kashyap, and Barath Raj Kandur Raja. What’s in the flow? exploiting temporal motion cues for unsupervised generic event boundary detection. In *WACV*, pages 6941–6950, 2024. 2, 3
- [13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4
- [14] Rui He, Yuanxi Sun, Youzeng Li, Zuwei Huang, Feng Hu, Xu Cheng, and Jie Tang. Masked autoencoders for generic event boundary detection cvpr’2022 kinetics-gebd challenge. *arXiv preprint arXiv:2206.08610*, 2022. 5
- [15] Ziyuan Huang, Shiwei Zhang, Liang Pan, Zhiwu Qing, Mingqian Tang, Ziwei Liu, and Marcelo H Ang Jr. Tada! temporally-adaptive convolutions for video understanding. In *ICLR*, 2022. 2
- [16] Hyolim Kang, Kyungmin Kim, Yumin Ko, and Seon Joo Kim. Cag-qil: Context-aware actionness grouping via q imitation learning for online temporal action localization. In *ICCV*, pages 13729–13738, 2021. 1, 3
- [17] Hyolim Kang, Jinwoo Kim, Taehyun Kim, and Seon Joo Kim. Uboco: Unsupervised boundary contrastive learning for generic event boundary detection. In *CVPR*, pages 20073–20082, 2022. 2, 4, 5, 6, 3
- [18] Hyolim Kang, Hanjung Kim, Joungbin An, Minsu Cho, and Seon Joo Kim. Soft-landing strategy for alleviating the task discrepancy problem in temporal action localization tasks. In *CVPR*, pages 6514–6523, 2023. 1
- [19] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 5
- [20] Lei Ke, Martin Danelljan, Henghui Ding, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Mask-free video instance segmentation. In *CVPR*, pages 22857–22866, 2023. 3
- [21] Young Hwi Kim, Hyolim Kang, and Seon Joo Kim. A sliding window scheme for online temporal action localization. In *ECCV*, pages 653–669. Springer, 2022. 3
- [22] Colin Lea, Austin Reiter, René Vidal, and Gregory D Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In *ECCV*, pages 36–52. Springer, 2016. 6, 2
- [23] Congcong Li, Xinyao Wang, Dexiang Hong, Yufei Wang, Libo Zhang, Tiejian Luo, and Longyin Wen. Structured context transformer for generic event boundary detection. *arXiv preprint arXiv:2206.02985*, 2022. 2, 3
- [24] Congcong Li, Xinyao Wang, Longyin Wen, Dexiang Hong, Tiejian Luo, and Libo Zhang. End-to-end compressed video representation learning for generic event boundary detection. In *CVPR*, pages 13967–13976, 2022. 2
- [25] Chuming Lin, Chengming Xu, Donghao Luo, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yanwei Fu. Learning salient boundary feature for anchor-free temporal action localization. In *CVPR*, pages 3320–3329, 2021. 2, 3
- [26] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *ICCV*, pages 3889–3898, 2019. 6, 2
- [27] Xiaolong Liu, Song Bai, and Xiang Bai. An empirical study of end-to-end temporal action detection. In *CVPR*, pages 20010–20019, 2022. 2
- [28] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, pages 3202–3211, 2022. 1
- [29] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

- [30] Gabriel A Radvansky and Jeffrey M Zacks. Event perception. *Wiley Interdiscip. Rev. Cogn. Sci.*, 2(6):608–620, 2011. 4
- [31] Ayush K Rai, Tarun Krishna, Julia Dietlmeier, Kevin McGuinness, Alan F Smeaton, and Noel E O’Connor. Motion aware self-supervision for generic event boundary detection. In *WACV*, pages 2728–2739, 2023. 2
- [32] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Intra-and inter-action understanding via temporal action parsing. In *CVPR*, pages 730–739, 2020. 2, 5, 3
- [33] Dingfeng Shi, Yujie Zhong, Qiong Cao, Lin Ma, Jia Li, and Dacheng Tao. Tridet: Temporal action detection with relative boundary modeling. In *CVPR*, pages 18857–18866, 2023. 2
- [34] Mike Zheng Shou, Stan Weixian Lei, Weiyao Wang, Deepti Ghadiyaram, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. In *ICCV*, pages 8075–8084, 2021. 1, 2, 3, 5, 6
- [35] Zheng Shou, Juntong Pan, Jonathan Chan, Kazuyuki Miyazawa, Hassan Mansour, Anthony Vetro, Xavier Giro-i Nieto, and Shih-Fu Chang. Online detection of action start in untrimmed, streaming videos. In *ECCV*, pages 534–551, 2018. 3
- [36] Haisheng Su, Weihao Gan, Wei Wu, Yu Qiao, and Junjie Yan. Bsn++: Complementary boundary regressor with scale-balanced relation modeling for temporal action proposal generation. In *AAAI*, pages 2602–2610, 2021. 1
- [37] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *CIKM*, pages 1441–1450, 2019. 4
- [38] Jing Tan, Yuhong Wang, Gangshan Wu, and Limin Wang. Temporal perceiver: A general architecture for arbitrary boundary detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10):12506–12520, 2023. 2, 3
- [39] Jiaqi Tang, Zhaoyang Liu, Chen Qian, Wayne Wu, and Limin Wang. Progressive attention on multi-level dense difference maps for generic event boundary detection. In *CVPR*, pages 3355–3364, 2022. 2, 5, 3
- [40] Song Tang, Chuang Li, Pu Zhang, and RongNian Tang. Swinlstm: Improving spatiotemporal prediction accuracy using swin transformer and lstm. In *ICCV*, pages 13470–13479, 2023. 4
- [41] Tuan N Tang, Jungin Park, Kwonyoung Kim, and Kwanghoon Sohn. Simon: A simple framework for online temporal action localization. *arXiv preprint arXiv:2211.04905*, 2022. 2, 5, 6
- [42] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 2
- [43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017. 4
- [44] Xiang Wang, Shiwei Zhang, Zhiwu Qing, Yuanjie Shao, Zhengrong Zuo, Changxin Gao, and Nong Sang. Oadtr: Online action detection with transformers. In *ICCV*, pages 7565–7575, 2021. 2, 5, 6
- [45] Xiao Wang, Jingen Liu, Tao Mei, and Jiebo Luo. Coseg: Cognitively inspired unsupervised generic event segmentation. *IEEE Trans. Neural Netw. Learn. Syst.*, 2023. 2, 5, 6, 3
- [46] Junfeng Wu, Yi Jiang, Song Bai, Wenqing Zhang, and Xiang Bai. Seqformer: Sequential transformer for video instance segmentation. In *ECCV*, pages 553–569. Springer, 2022. 4
- [47] Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *CVPR*, pages 10156–10165, 2020. 2
- [48] Mingze Xu, Yuanjun Xiong, Hao Chen, Xinyu Li, Wei Xia, Zhuowen Tu, and Stefano Soatto. Long short-term transformer for online action detection. *NeurIPS*, 34:1086–1099, 2021. 3
- [49] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *ICCV*, pages 5188–5197, 2019. 3
- [50] Jeffrey M Zacks and Khena M Swallow. Event segmentation. *Curr. Dir. Psychol. Sci.*, 16(2):80–84, 2007. 1, 2
- [51] Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In *ECCV*, pages 492–510. Springer, 2022. 1, 2, 3
- [52] Libo Zhang, Xin Gu, Congcong Li, Tiejian Luo, and Heng Fan. Local compressed video stream learning for generic event boundary detection. *IJCV*, 132(4):1187–1204, 2024. 2, 3
- [53] Chen Zhao, Shuming Liu, Karttikeya Mangalam, and Bernard Ghanem. Re2tal: Rewiring pretrained video backbones for reversible temporal action localization. In *CVPR*, pages 10637–10647, 2023. 2
- [54] Yue Zhao and Philipp Krähenbühl. Real-time online video detection with temporal smoothing transformers. In *ECCV*, pages 485–502. Springer, 2022. 1, 2, 3, 5, 6, 8
- [55] Ziwei Zheng, Lijun He, Le Yang, and Fan Li. Fine-grained dynamic network for generic event boundary detection. In *ECCV*, pages 107–123. Springer, 2024. 2, 3
- [56] Ziwei Zheng, Zechuan Zhang, Yulin Wang, Shiji Song, Gao Huang, and Le Yang. Rethinking the architecture design for efficient generic event boundary detection. In *ACMMM*, pages 1215–1224, 2024. 2, 3

Online Generic Event Boundary Detection

Supplementary Material

Table 8. **Ablation study of Online Boundary Discriminator.** Each number denotes an Avg. F1 score for the range of τ and Δ . Bold number denotes the best Avg. F1 among the given same size of queue.

τ	Size of Queue Δ				
	12	15	18	21	24
1.0	0.705	0.705	0.741	0.748	0.751
1.5	0.701	0.701	0.743	0.748	0.747
2.0	0.715	0.715	0.740	0.735	0.726
2.5	0.731	0.728	0.716	0.701	0.684
3.0	0.711	0.691	0.670	0.649	0.629

A. Ablation Study on Both τ and δ in OBD

In Table 8, we conduct an in-depth analysis on the interaction between the threshold τ and the queue size Δ for the Online Boundary Discriminator (OBD). This table highlights the effects of varying these two parameters on the average F1 (Avg. F1) score in Kinetics-GEBD dataset [34]. We observe that for a given queue size Δ , increasing the threshold τ initially leads to improvements in performance up to a certain point, after which further increases in τ lead to a decline in the Avg. F1 score. For instance, when the queue size is fixed at $\Delta = 18$, the peak performance is achieved at $\tau = 1.5$, with an Avg. F1 score of 0.743. Increasing the threshold means selecting more severe outliers compared to the past errors stored in the OBD. Thus, setting a criterion that is either too strict or not would naturally result in a decline in overall performance. We have determined τ as 1.5 throughout the entire experiment, since it demonstrates satisfactory performance as shown in the Table 8.

Additionally, we can observe that performance gets better with lower τ values when Δ increases. For example, at a queue size of $\Delta = 24$, the highest F1 score is 0.751, which occurs at the lowest examined threshold of $\tau = 1.0$. This trend suggests that larger queues are better with lower thresholds, potentially due to the greater amount of past errors available in OBD queue when determining event boundaries. We choose a queue size of $\Delta = 21$ and a threshold of $\tau = 1.5$, where the model achieves its optimal performance with an Avg. F1 score of 0.748.

B. Further Experiments on K in REST Loss

The Regional EST (REST) loss is a core component in training our Consistent Event Anticipator (CEA), designed to enhance the model’s ability to detect subtle changes at event boundaries. The parameter K determines the size of the temporal region considered in the REST loss calcula-

Table 9. **Ablation study of K in REST loss.** Adjusting the range of REST loss in training CEA.

K	3	5	7	9	11	13	15	17	19
Avg F1	0.724	0.733	0.743	0.748	0.756	0.756	0.754	0.749	0.746

Table 10. **Comparison of different lengths, Avg F1 scores, and VRAM usage.** We denote the highest Avg F1 in **bold**.

Length	Avg F1	VRAM (GB)
4	0.728	5.2
8	(Ours) 0.748	9.0
16	0.742	14.8
32	0.745	27.9

tion, controlling the range of frames that influences the loss computation. To better understand the impact of this parameter, we conducted additional experiments varying the size of K , with results presented in Table 9. These experiments reveal a clear trend in model performance as K changes. The Avg. F1 score shows a consistent increase as K grows from 3 to 11, indicating that larger temporal context benefits the model’s ability to detect event boundaries. This improvement can be attributed to the model’s enhanced capacity to capture longer-range dependencies and more complex temporal patterns within the video sequences.

Interestingly, our experimental result shows that the model’s performance peaks when K is set to 11 or 13, with both values yielding an Avg. F1 of 0.756. However, we observe a decline in performance for K values beyond 13, suggesting that excessively large temporal regions may introduce noise or irrelevant information into the loss calculation. Despite the highest performance at $K = 11$ and 13, we opted to use $K = 9$ for all experiments reported in the main manuscript. This decision was primarily due to practical considerations, considering the trade-off between model performance and computational resources. Larger K values require more GPU VRAM during training, which can limit batch sizes or necessitates more powerful hardware.

C. Ablation on Length L

The choice of input video sequence length impacts both the performance and computational efficiency of our model. A longer input sequence provides more temporal context, potentially improving boundary detection accuracy but at the cost of increased VRAM consumption and inference time. Conversely, shorter sequences are computationally efficient but may lack sufficient context for detecting subtle event transitions.

Table 11. **Quantitative comparison with additional offline methods.** In addition to the offline GEBD methods presented in Table 2 of our original manuscript, we include additional results from more recent offline approaches to highlight the robustness of our model, even as an online method. Note that we report the performance of the models in an offline setting from their original literature. Also, we indicate the highest F1 score with **bold** for each dataset.

Dataset	Setting	Supervision	Rel. Dis. threshold	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	Avg
Kinetics-GBD	Offline	Supervised	BMN [26]	0.186	0.204	0.213	0.220	0.226	0.230	0.233	0.237	0.239	0.241	0.223
			BMN-StartEnd [34]	0.491	0.589	0.627	0.648	0.660	0.668	0.674	0.678	0.681	0.683	0.640
			TCN-TAPOS [34]	0.464	0.560	0.602	0.628	0.645	0.659	0.669	0.676	0.682	0.687	0.627
			TCN [22]	0.588	0.657	0.679	0.691	0.698	0.703	0.706	0.708	0.710	0.712	0.685
			PC [34]	0.625	0.758	0.804	0.829	0.844	0.853	0.859	0.864	0.867	0.870	0.817
			Temporal Perceiver [38]	0.748	0.828	0.852	0.866	0.874	0.879	0.883	0.887	0.890	0.892	0.860
			SBoCo-Res50 [17]	0.732	-	-	-	-	-	-	-	-	-	0.866
			DDM-Net [39]	0.764	0.843	0.866	0.880	0.887	0.892	0.895	0.898	0.900	0.902	0.873
			SC-Transformer [23]	0.777	0.849	0.873	0.886	0.895	0.900	0.904	0.907	0.909	0.911	0.881
			EfficientGEBD [56]	0.783	0.851	-	-	-	0.901	-	-	-	0.913	0.883
		Unsupervised	LCVSL [52]	0.768	0.848	0.872	0.885	0.892	0.896	0.899	0.901	0.903	0.906	0.877
			DyBDet [55]	0.796	0.858	0.880	0.893	0.901	0.907	0.911	0.915	0.917	0.919	0.890
			SceneDetect [34]	0.275	0.300	0.312	0.319	0.324	0.327	0.330	0.332	0.334	0.335	0.318
			PA-Random [34]	0.336	0.435	0.484	0.512	0.529	0.541	0.548	0.554	0.558	0.561	0.506
			PA [34]	0.396	0.488	0.520	0.534	0.544	0.550	0.555	0.558	0.561	0.564	0.527
			CoSeg [45]	0.656	0.758	0.783	0.794	0.799	0.803	0.804	0.806	0.807	0.809	0.782
			UBoCo-Res50 [17]	0.703	-	-	-	-	-	-	-	-	-	0.867
			FlowGEBD [12]	0.713	0.828	0.850	0.858	0.862	0.864	0.866	0.867	0.868	0.869	0.845
	Online	Supervised	ESTimator (Ours)	0.620	0.687	0.724	0.746	0.762	0.774	0.782	0.789	0.795	0.799	0.748
TAPOS	Offline	Supervised	ISBA [34]	0.106	0.170	0.227	0.265	0.298	0.326	0.348	0.348	0.348	0.348	0.330
			TCN [34]	0.237	0.312	0.331	0.339	0.342	0.344	0.347	0.348	0.348	0.348	0.330
			CTM [34]	0.244	0.312	0.336	0.351	0.361	0.369	0.374	0.381	0.383	0.385	0.350
			TransParser [22]	0.289	0.381	0.435	0.475	0.500	0.514	0.527	0.534	0.540	0.545	0.474
			PC [34]	0.522	0.595	0.628	0.646	0.659	0.665	0.671	0.676	0.679	0.683	0.642
			DDM-Net [39]	0.604	0.681	0.715	0.735	0.747	0.753	0.757	0.760	0.763	0.767	0.728
			Temporal Perceiver [38]	0.552	0.663	0.713	0.738	0.757	0.765	0.774	0.779	0.784	0.788	0.732
			SC-Transformer [23]	0.618	0.694	0.728	0.749	0.761	0.767	0.771	0.774	0.777	0.780	0.742
			EfficientGEBD [56]	0.631	0.705	-	-	-	0.774	-	-	-	0.786	0.748
			LCVSL [52]	0.618	0.694	0.728	0.749	0.761	0.767	0.771	0.774	0.777	0.780	0.742
			DyBDet [55]	0.625	0.701	0.734	0.756	0.767	0.772	0.775	0.779	0.781	0.784	0.747
		Unsupervised	SceneDetect [34]	0.035	0.045	0.047	0.051	0.053	0.054	0.055	0.056	0.057	0.058	0.051
			PA-Random [34]	0.158	0.233	0.273	0.310	0.331	0.347	0.357	0.369	0.376	0.384	0.314
			PA [34]	0.360	0.459	0.507	0.543	0.567	0.579	0.592	0.601	0.609	0.615	0.543
			FlowGEBD [12]	0.375	0.502	0.569	0.624	0.658	0.677	0.695	0.703	0.711	0.717	0.623
	Online	Supervised	ESTimator (Ours)	0.394	0.455	0.499	0.532	0.558	0.578	0.594	0.608	0.619	0.629	0.547

To achieve a balance between performance and efficiency, we set the input length to an optimal value based on empirical results. As shown in Table 10, we compare different sequence lengths in terms of Avg F1 score and VRAM usage. Our selected input length achieves the highest Avg F1 score while maintaining a reasonable VRAM footprint, making it suitable for real-time processing.

Our OBD is designed to dynamically adapt to recent boundary patterns, reducing false positives during frequent changes while maintaining sensitivity in stable periods. This design aligns with human perception, as studies suggest that when individuals are exposed to rapidly changing visuals, they naturally adjust their threshold for identifying meaningful event boundaries [11]. The ability to incorporate past outliers ensures that the model remains adaptable to varying event structures without excessive desensitization to new transitions.

Table 12. **Ablation on batch-wise weighted loss.**

Batch-wise loss	Avg F1
$\times$	0.743
$\checkmark$	0.748

These findings reinforce the necessity of including outliers in the queue to maintain robust event boundary detection, making our approach both computationally effective and cognitively plausible.

D. Additional Offline GEBD performance table

We further report the performance of models developed and evaluated under an offline setting in Table 11. Compared

Table 13. **Quantitative comparison for generalization ability.** Results on Youtube-INRIA-Instructional dataset with online and offline baselines.

Online	Method	Pretrained	Precision@0.05	Recall@0.05	F1@0.05
X	U-Net	INRIA	-	-	0.299
	CoSeg [41]	INRIA	0.467	0.633	0.537
O	TeSTra – BC	Kinetics-GEBD	0.181	0.748	0.291
	Sim-On – BC	Kinetics-GEBD	0.099	0.068	0.080
	OadTR – BC	Kinetics-GEBD	0.348	0.526	0.419
	MiniROAD - BC	Kinetics-GEBD	0.209	0.572	0.306
	Ours	Kinetics-GEBD	0.411	0.666	0.508

to the Table 2 in our main manuscript, Table 11 additionally include Temporal Perceiver [38], SBoCo-Res50 [17], DDM-Net [39], SC-Transformer [23], UBoCo [17], Efficient-GEBD [56], LCVSL [52], DyBDet [55] and FlowGEBD [12] for the Kinetics-GEBD dataset. For the TAPOS dataset, we have additionally included DDM-Net, Temporal Perceiver, SC-Transformer, Efficient-GEBD [56], LCVSL [52], DyBDet and FlowGEBD [12] as UBoCo do not report performance for this dataset.

E. Ablation on Batch-wise Weighted Loss

Table 12 presents the Avg. F1 score on the Kinetics-GEBD dataset, evaluating the impact of batch-wise weighted loss in our model. This technique addresses the imbalance between boundary and non-boundary frames in the training data, a common challenge in event boundary detection tasks. By dynamically adjusting the importance of samples within a single batch during training, the batch-wise weighted loss aims to improve the model’s sensitivity to boundary frames without manual hyper-parameter tuning.

The results indicate that incorporating batch-wise weighted loss yields a 0.5%p increase in the Avg. F1 score. This improvement may seem trivial, but considering the sensitivity of detecting generic event boundaries, we conjecture that batch-wise weighting is showing noticeable improvement in accuracy.

F. Zero-shot Ability of Our Framework

To further demonstrate the generalization capability of our framework, we evaluate our framework on the challenging YouTube-INRIA-Instructional dataset [1] (Table 13), which was used in [45] and consists of long-form, multi-minute instructional videos—markedly different in nature from Kinetics-GEBD. Without any additional finetuning, our model pretrained solely on Kinetics-GEBD achieves an F1@0.05 score of 0.508. This result is competitive with, or even superior to, existing offline methods, and it consistently outperforms all online baselines. These results highlight the strong zero-shot generalization ability of our model to previously unseen, complex video domains.

G. Additional Details on Computational Cost

In Table 5 of the main manuscript, we analyze the real-time performance of our proposing model, focusing on its inference speed (*i.e.* FPS). For completeness, we provide additional real-time metrics including computational cost details (*e.g.*, GFLOPs and memory usage) in Table 14, highlighting the efficiency of our method in online scenario. As showcased in the Table 14, our model achieves best performance despite having compatible number of GFLOPs and parameters compared to the most efficient baselines (*i.e.*, Sim-On-BC, MiniROAD-BC), demonstrating the effectiveness.

H. Additional Qualitative Result

We illustrate more qualitative results of our model compared to one of baselines (TeSTra-BC [54]), on both Kinetics-GEBD and TAPOS [32] datasets. In Figure 5, we present two cases of abrupt scene changes (*i.e.*, first and second row) and two cases of subtle changes (*i.e.*, third and fourth row) in Kinetics-GEBD dataset.

The first row shows a distinct transition such as shot changes between events in a video. In this straightforward scenario, both the baseline and our method yield results that are close to the ground truth. However, the error plot of our method for each frame shows sharp peaks, distinctively indicating the boundary locations, in contrast to the baseline’s, which presents a nearly flat distribution. In the second row, there are changes of scene not only at event boundaries but also within each event. While TeSTra-BC fails to recognize the semantic continuity at the first event of the video and raises numerous false alarms, our framework recognizes the boundaries successfully. The third and fourth example present cases where the transition of events is subtle, requiring a deeper understanding of granular details to detect event boundaries. Our model also outperforms the baseline in identifying event boundaries.

In Figure 6, we present a comparison between TeSTra-BC and our framework on the TAPOS dataset. As mentioned in our main manuscript, the TAPOS dataset consists of Olympic sport videos annotated with 21 action classes, where each action is further divided into multiple sub-actions. Since these sub-actions are re-purposed as a single

Table 14. **Comparison of real-time performance with computational cost.** Note that **bold** refers to the best and underline refers to the second best.

Method	# of param.	GFLOPs ↓	VRAM (MB) ↓	FPS ↑	Avg. F1 ↑
TeSTra – BC	48.73M	17.0	354	72.5	0.557
Sim-On – BC	24.70M	8.2	134	76.3	0.618
OadTR – BC	97.10M	13.0	385	48.9	0.558
MiniROAD - BC	37.15M	8.2	134	99.8	<u>0.681</u>
Ours	42.41M	<u>10.3</u>	<u>228</u>	<u>96.3</u>	0.748

*All experiments were conducted on a single NVIDIA RTX A6000 GPU.

event in our experiment, the semantic changes between sub-actions within the single video tend to be subtle. As shown in Figure 6, TeSTra-BC fails to detect event boundaries in all four cases, particularly failing to detect any boundaries in the third and fourth cases. In contrast, our framework successfully detects the subtle semantic changes occurring at event boundaries in all videos.

I. Limitation and Social Impact

Although the Kinetics-GEBD and TAPOS dataset are the only datasets available for testing the GEBD task, they consist exclusively of sports or exercise-related videos. In this context, OBD, which introduces a novel criterion for defining event boundaries, may exhibit bias toward sports or exercise contexts. To ensure robust performance across a diverse range of domains, it may be necessary to construct a variety of datasets for GEBD and perform a tuning of corresponding parameters (*e.g.*, Δ , τ).

Since the On-GEBD solver is able to process diverse long-form videos in real time, it has the potential to impact fields that require continuous monitoring and rapid analysis within the previously unobserved video streams such as public safety and surveillance.

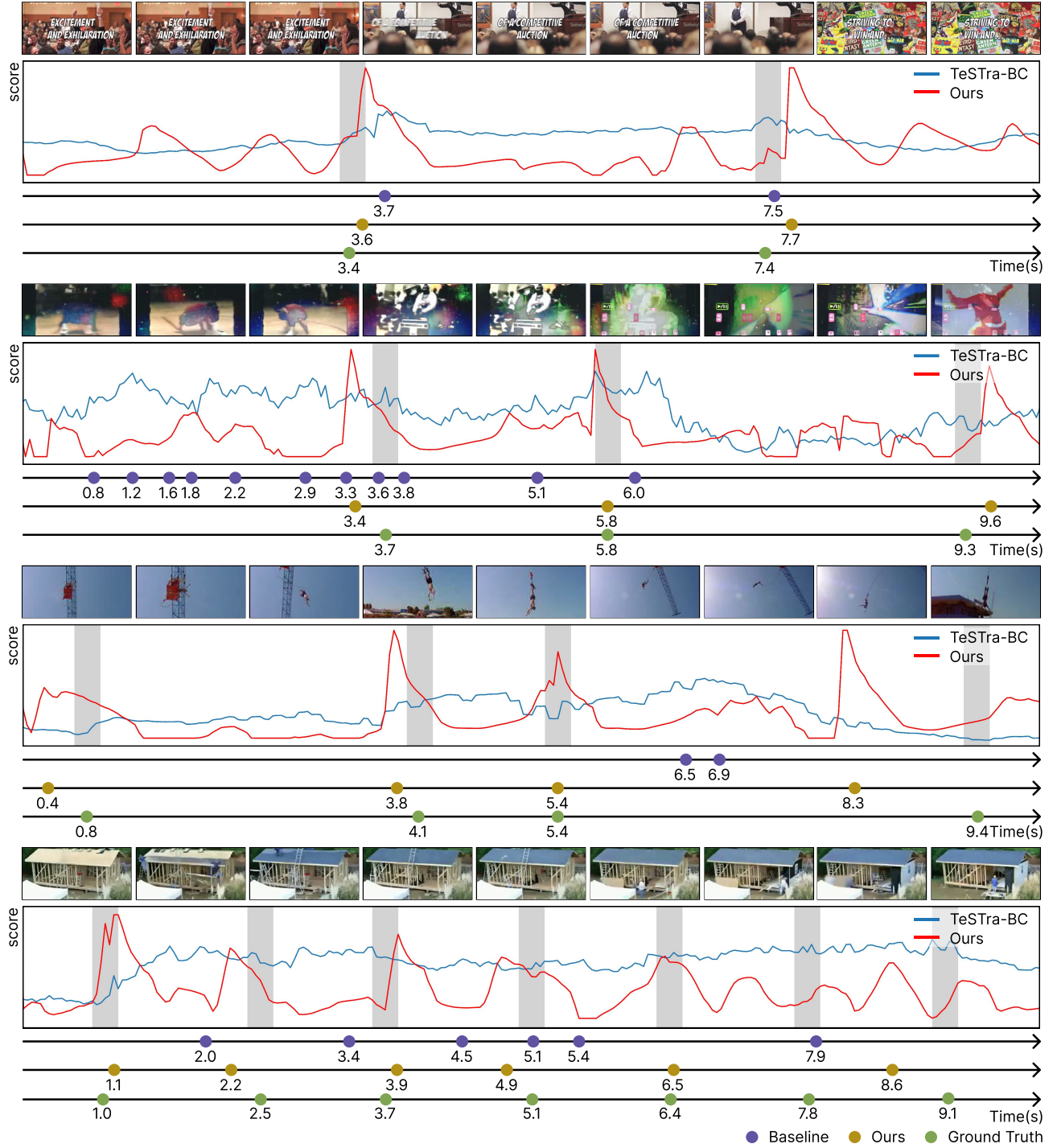


Figure 5. **Additional qualitative result on Kinetics-GEBD dataset.** Comparison between our proposed framework and the baseline (TeSTra-BC [54]).

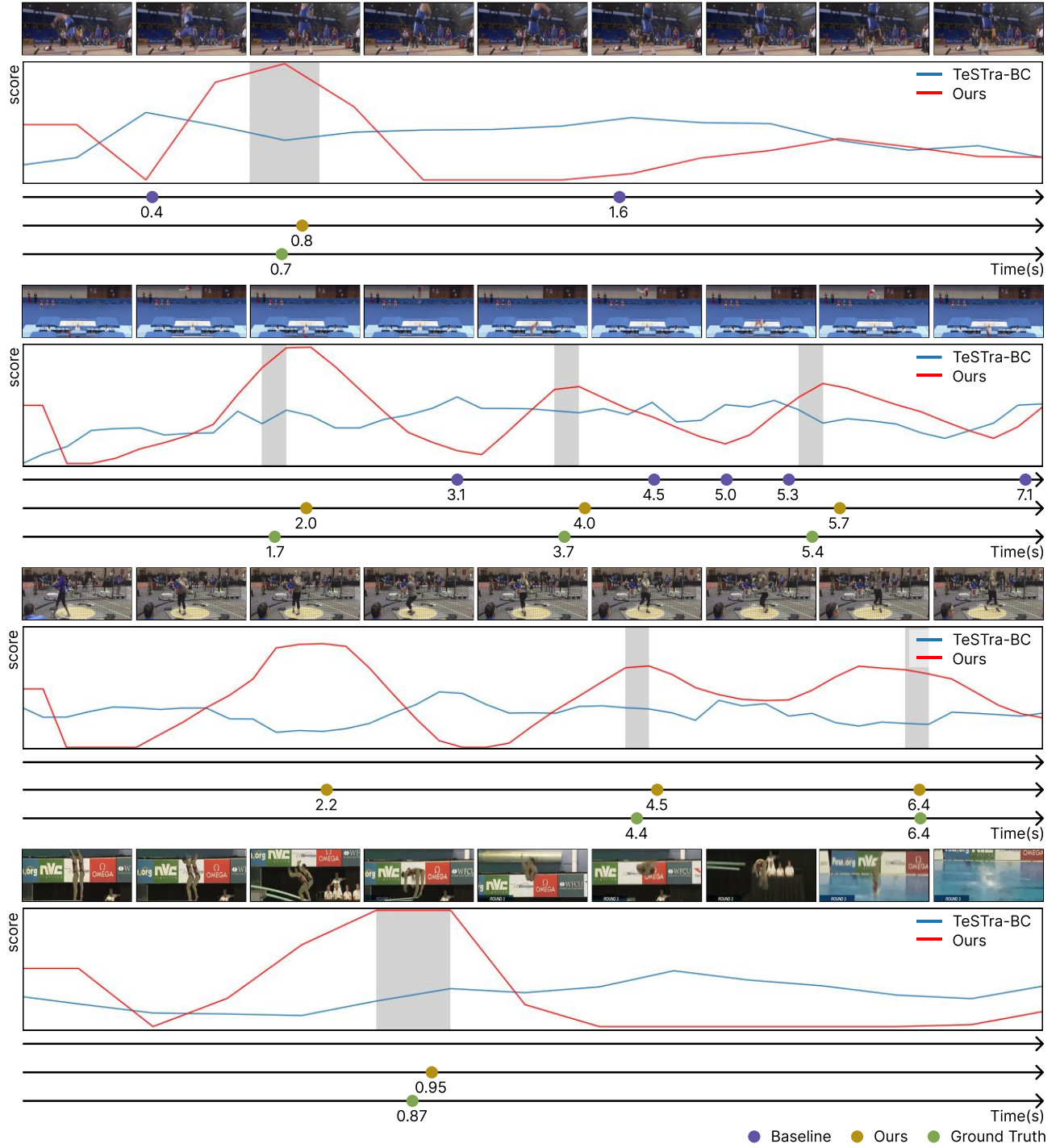


Figure 6. **Additional qualitative result on TAPOS dataset.** Comparison between our proposed framework and the baseline (TeSTra-BC [54]).