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 determied 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 calculation, 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.

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

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-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
			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
			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**
		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	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
✗	0.743
✓	**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 to the Table 2 in our main manuscript, Table 11 additionally include Temporal Perceiver [38], SBoCo-Res50 [17],

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**

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**	0.681
Ours	42.41M	10.3	228	96.3	**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.

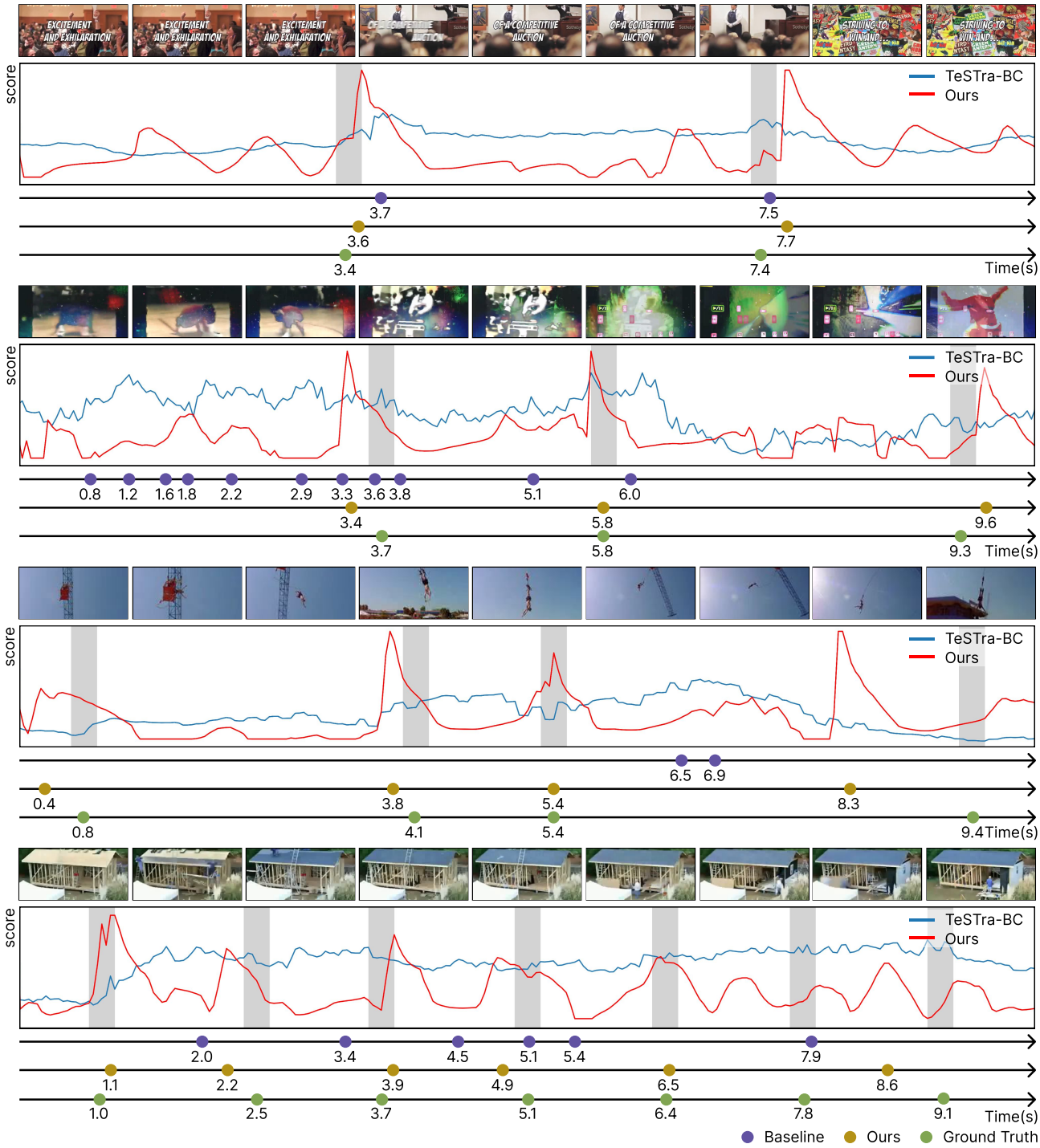

Figure 5. **Additional qualitative result on Kinetics-GEBD dataset.** Comparison between our proposed framework and the baseline (TeSTra-BC [54]).

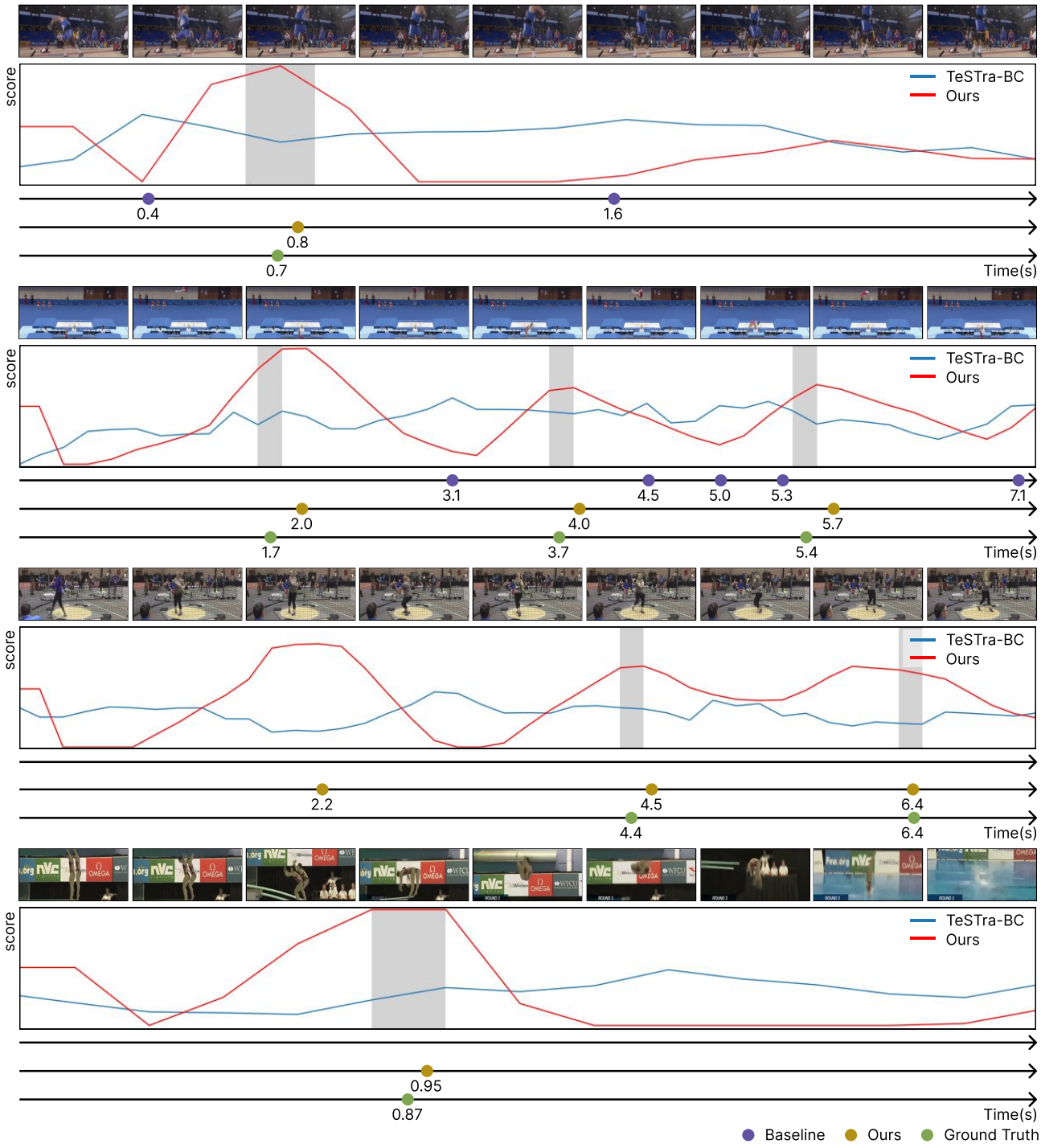

Figure 6. **Additional qualitative result on TAPOS dataset.** Comparison between our proposed framework and the baseline (TeSTra-BC [54]).