

# UBoCo : Unsupervised Boundary Contrastive Learning for Generic Event Boundary Detection

Hyolim Kang \*, Jinwoo Kim \*, Taehyun Kim , Seon Joo Kim

Yonsei University

{hyolimkang, jinwoo-kim, kimth0101, seonjookim}@yonsei.ac.kr

## Abstract

*Generic Event Boundary Detection (GEBD) is a newly suggested video understanding task that aims to find one level deeper semantic boundaries of events. Bridging the gap between natural human perception and video understanding, it has various potential applications, including interpretable and semantically valid video parsing. Still at an early development stage, existing GEBD solvers are simple extensions of relevant video understanding tasks, disregarding GEBD’s distinctive characteristics. In this paper, we propose a novel framework for unsupervised/supervised GEBD, by using the Temporal Self-similarity Matrix (TSM) as the video representation. The new Recursive TSM Parsing (RTP) algorithm exploits local diagonal patterns in TSM to detect boundaries, and it is combined with the Boundary Contrastive (BoCo) loss to train our encoder to generate more informative TSMs. Our framework can be applied to both unsupervised and supervised settings, with both achieving state-of-the-art performance by a huge margin in GEBD benchmark. Especially, our unsupervised method outperforms the previous state-of-the-art “supervised” model, implying its exceptional efficacy.*

## 1. Introduction

With the proliferation of video platforms, video understanding tasks are drawing substantial attention in computer vision community. The prevailing convention for video processing [3, 15, 16, 21, 28] is still dividing the whole video into short non-overlapping snippets with a fixed duration, which neglects the semantic continuity of the video. On the other hand, cognitive scientists have observed that human senses the visual stream as a set of events [39], which alludes that there is room for research to find out a video parsing method

that preserves semantic validity and interpretability of video snippets.

From this perspective, Generic Event Boundary Detection (GEBD) [37] can be seen as a new attempt to interconnect human perception mechanisms to video understanding. The goal of GEBD is to pinpoint the content change moments in which humans perceive them as event boundaries. To reflect human perception, labels for the task are annotated following the instruction of finding boundaries of *one level deeper* granularity compared to the video-level event, regardless of action classes. This characteristic differentiates GEBD from the previous video localization tasks [42] in the sense that the labels are only given by natural human perception, not the predefined action classes.

Recently released Kinetics-GEBD [37] is the first dataset for the GEBD. The distinctive point in the dataset is that the event boundary labels are annotated by 5 different annotators, making the dataset convey the subjectivity of human perception. Besides, various baseline methods for the GEBD task were also included in [37]. Based on the similarity to Temporal Action Localization (TAL), many of the suggested GEBD methods were extensions of previous TAL works [26, 27], while several unsupervised methods exploited shot detection methods and event segmentation theory [25, 36, 43]. Detailed explanations about baseline GEBD methods can be found in Section 2.

Nevertheless, straightforward adoptions of existing methods have a clear limitation in that they directly use pre-trained features to predict boundary points. As the features are extracted from the classification-pretrained networks like ResNet-50 [19], they inevitably focus on class-specific or object-centric information. However, many event boundaries, especially those that are hard to detect, do not entail scene change (e.g. *Stand-by* frame and *Running* frame in Figure 1 share most of the scene information). For capturing event boundaries, what really matters is the *relationship* between adjacent frames, implying the need for a new

\*equal contribution, ordered by surname

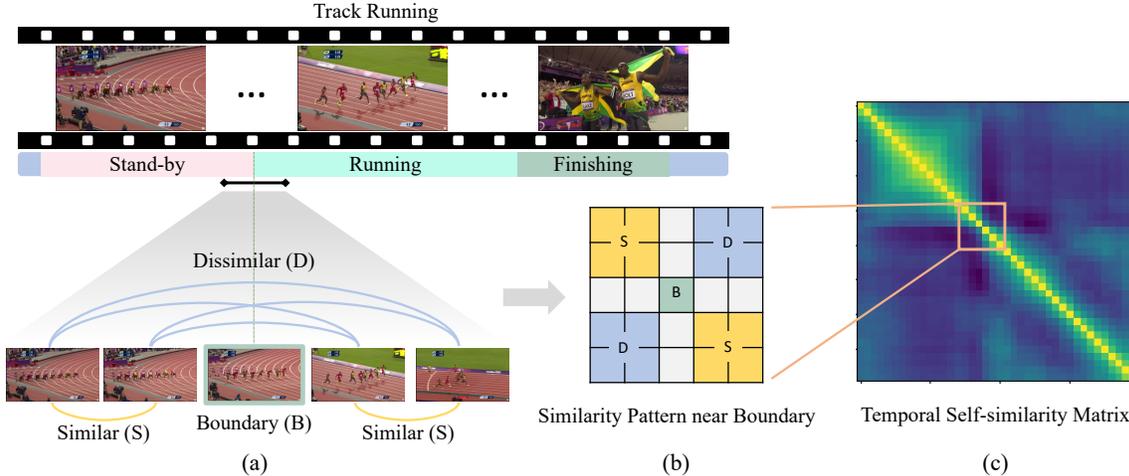


Figure 1. GEBC is a task of finding boundaries of one level deeper granularity compared to the video level event. (a) The relationship between frames showing that the *local* similarities between adjacent frames are maintained except near the event boundary. (b) Similarity pattern near a event boundary  $B$  where the yellow region  $S$  indicates high similarity score and the blue region  $D$  indicates low similarity score. (c) Temporal Self-similarity Matrix (TSM) represents the pairwise self-similarity scores between video frames. Similar patterns are observed in event boundaries, and we can detect boundary frames by mining this pattern from the TSM.

method to properly exploit it.

In this paper, we introduce a novel method to discover generic event boundaries, which can be put into both *unsupervised* and *supervised* settings. Our main intuition comes from observing the self-similarity of a video, visualized as Temporal Self-similarity Matrix (TSM). While TSM has been considered as a useful tool to analyze *periodic* videos due to its robustness against noise [1, 2, 14, 34], we found that TSM’s potential is not limited to periodic videos, but can also be extended to analyzing *non-periodic* videos if we focus on its local diagonal patterns. To be specific, we can exploit TSM’s robustness by taking it as an information bottleneck for the GEBC solver, making it perform well on unseen scenes, objects, and even action classes [14].

Figure 1 briefly illustrates our observation. For a sequence of events in a given video, there is a semantic inconsistency at the event boundary, resulting in the similarity-break near the boundary point. As TSM depicts self-similarity scores between video frames, this similarity-break brings distinctive patterns (Figure 1 (c)) on TSM, which can be a meaningful cue when it comes to detecting event boundaries. Hence, we exploit TSM as the *final* representation of the given video and devise a novel method to detect event boundaries, namely Recursive TSM Parsing (RTP). Paired with our Boundary Contrastive loss (BoCo loss), RTP can be extended to *Unsupervised* Boundary Contrastive (UBoCo) learning, a fully label-free training framework for event boundary detection.

Going one step further, we also propose *Supervised* Boundary Contrastive (**SBoCo**) learning approach for the GEBC task, which utilizes TSM as an interpretable *inter-*

*mediate* representation. Unlike UBoCo that uses an algorithmic method to parse TSM, this supervised approach has TSM decoder, which is a standard neural network. By merging binary cross-entropy (BCE) and BoCo loss, our supervised approach achieved the state-of-the-art performance in recent official GEBC challenge<sup>1</sup>.

To summarize, the main contribution of the paper is as follows:

- We discovered that the properties of Temporal Self-Similarity Matrix (TSM) matches very well with the Generic Event Boundary Detection (GEBC) task, and propose to use TSM as the representation of the video to solve GEBC.
- Taking advantage of TSM’s distinctive boundary patterns, we propose Recursive TSM Parsing (RTP) algorithm, which is a divide-and-conquer approach for detecting event boundaries.
- Unsupervised framework for GEBC is introduced by combining RTP and the new Boundary Contrastive (BoCo) loss. Using the BoCo loss, the video encoder can be trained without labels and generate more distinctive TSMs. Our unsupervised framework outperforms not only previous unsupervised methods, but also supervised methods.
- Our framework can be easily extended to the supervised setting by adding a decoder and achieve the state-of-the-art performance by a large margin (16.2%).

<sup>1</sup>CVPR’21 LOnG-form VidEo Understanding (LOVEU) Kinetics-GEBC challenge

## 2. Related Work

### 2.1. Generic Event Boundary Detection

Generic Event Boundary Detection (GEBD) [37] is a newly introduced video understanding task that aims to spot event boundaries that coincide with human perception. GEBD shares an important trait with the popular video event detection task, Temporal Action Localization (TAL) in that the model should be aware of the action happening in the video. Among many TAL methods, BMN [27] has the intermediate stage of calculating the probability of the start and the end point of an action instance, making its extension to GEBD more feasible. Therefore, [37] introduced a method called BMN-StartEnd, which treated BMN model’s intermediate action start-end detection results as event boundaries.

Apart from extending TAL, better performance can be achieved by treating the GEBD task as a framewise binary classification (boundary or not). For the network architecture, temporal modeling using TCN [26, 29] has been tested, but a simple linear classifier with concatenated average feature (denoted as PC in [37]) resulted in the best performance.

For unsupervised GEBD, a straightforward method is to utilize a previous shot boundary detector<sup>2</sup>. However, generic event boundaries consist of various kind of event boundaries including change of action, subject, and environment, implying that only a small portion of event boundaries can be detected with the shot boundary approach. Thus, [37] devised a novel unsupervised GEBD solver exploiting the fact that predictability is the main factor in human’s event perception [25]. Among many suggested unsupervised methods, the PA (PredictAbility) method resulted in the best performance.

Except for the PA method, many of the suggested GEBD approaches are just straightforward extensions of previous methods that targeted other video tasks, raising the need for a GEBD-specialized solution. Focusing on the GEBD’s distinctive characteristics, we suggest a novel method that exploits TSM representation, which shows a unique pattern near the boundary.

### 2.2. Temporal Self-similarity Matrix

With the increasing popularity of using self-attention for its benefits [4, 12, 13, 40], Temporal Self-similarity Matrix (TSM) has also been getting much attention lately as an interpretable intermediate representation for video. From a given video with  $L$  frames, each value at position  $(i, j)$  in TSM is calculated using cosine or L2-distance similarity between frame  $i$  and  $j$ , resulting in an  $L \times L$  matrix. As TSM represents the similarity between different frames

<sup>2</sup><https://github.com/Breakthrough/PySceneDetect>

in a video, it is often used for the task of repetition counting [14, 34], gait recognition [1, 2], and language-video localization [32]. Furthermore, as TSM effectively reflects temporal relationships between features, some works that require a general video representation like action classification [5], and representation learning [24] also utilize TSM as the intermediate representation. As the local temporal relationship is the key feature for the event boundary detection, our method also utilizes TSM as the video representation, enhancing the overall performance compared to previous methods.

### 2.3. Contrastive representation learning

Contrastive learning, with its generality and simplicity, is gaining increasing attention in computer vision society. Its main idea is to attract semantically matching samples closer and repel mismatching ones. Note that the method of determining whether a pair is semantically matching or not is not fixed in contrastive learning. Thus, while many recent works [7–9, 17, 18] combined contrastive learning with data augmentation and treated it as a self-supervised approach, some works extended contrastive learning to standard supervised learning [10, 23]. As our boundary contrastive loss utilizes (pseudo-) boundary labels for contrastive learning, it has a close relationship with the supervised contrastive learning.

Furthermore, there are various works on contrastive video representation learning, which viewed contrastive learning from temporal [33], spatio-temporal [35], and temporal-equivariant [20] perspectives. Especially, [6] applied self-supervised contrastive pretext task to learn appropriate feature representation, proving its effectiveness on detecting shot frames. However, it still needs *downstream* task training (supervised learning) to get final results, while our approach directly yields event boundaries.

## 3. Proposed Method

Given a video with  $L$  frames, GEBD solver returns a list containing event boundary frame indices. In this section, we introduce our novel unsupervised/supervised GEBD solver, which makes use of TSM in the intermediate stage.

### 3.1. Overview

Figure 2 shows an overview of our Unsupervised Boundary Contrastive learning framework (UBoCo). In the first stage of the procedure, framewise features are extracted from the raw video frames, using a neural feature extractor. The feature extractor consists of a pretrained frame encoder (ImageNet pretrained ResNet50 [19]) and an extra encoder. Weights are fixed for the pretrained encoder, while the custom encoder’s weights are trainable.

With the extracted feature, self-similarities between the frames are computed, forming Temporal Self-similarity

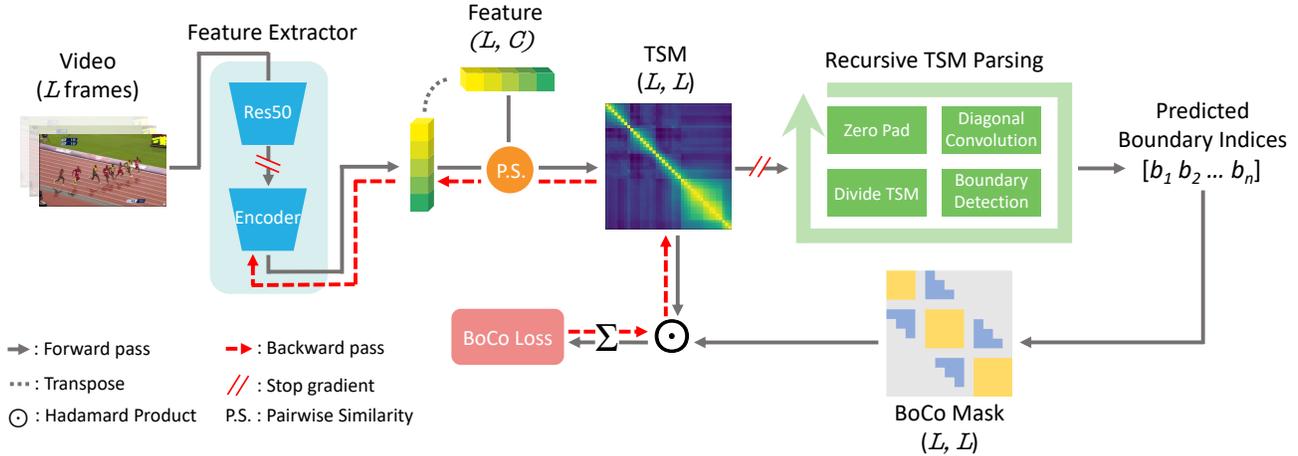


Figure 2. Overview of the Unsupervised Boundary Contrastive (UBoCo) learning framework for GEBD.

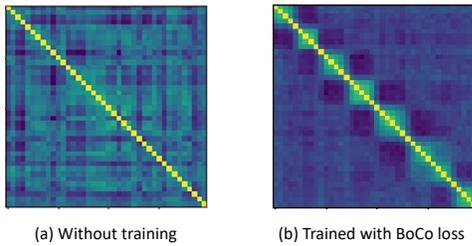


Figure 3. (a) Noisy TSM can be observed at the early stage of training. (b) As training proceeds, the TSM is getting sharper, showing distinctive boundary patterns.

Matrix (TSM). Given a TSM, Recursive TSM Parsing (RTP) algorithm (Section 3.2.1) produces boundary indices prediction. Considering this prediction as a pseudo-label, the encoder can be trained with the standard gradient-descent algorithm using the BoCo loss (Section 3.2.2), which enriches the encoder to produce boundary-sensitive features. Note that gradients directly flow to the TSM, by-passing the non-differentiable RTP pass.

Our concept of pseudo-label framework resembles the popular k-means clustering algorithm [30]. In k-means algorithm, the mean values of clustered vectors in the previous stage become the new criterion for the current clustering stage. The centroids of k-means clustering are analogous to pseudo-labels of our method in the sense that the current training step is conducted based on the result of the previous step.

In the early stage of training, TSM is not perfectly discriminative due to undertrained feature encoder (Figure 3). At this stage, only obvious boundaries that involve drastic visual changes can be detected by the RTP algorithm. With more training, the BoCo loss enables the feature encoder

to produce more robust TSM (Figure 3), making the discriminative power of TSM stronger. As a better feature encoder generates better quality pseudo-labels, the BoCo loss also becomes more powerful as the training proceeds. This progressive improvement is the key property of our UBoCo framework, and its effect on the overall performance will be shown in the Experiments Section.

## 3.2. Unsupervised Boundary Contrastive Learning

### 3.2.1 Recursive TSM Parsing

With the intuition illustrated in Figure 1, we devised a novel method called Recursive TSM Parsing (RTP), to detect event boundaries from a given TSM. As a divide-and-conquer approach, RTP algorithm takes TSM as its input and yields boundary indices in a *recursive* manner as shown in Figure 4.

First, the input TSM is zero-padded to make convolution operation to be applicable on corner elements (Figure 4 (a)). Then, diagonal elements of TSM are convolved with the “contrastive kernel”, which is a materialization of the boundary pattern in Figure 1. After the diagonal convolution with the contrastive kernel, scalar values that represent boundariness are produced (Figure 4 (b)). A higher boundary score means that the local TSM pattern matches well with the contrastive kernel, indicating a higher probability of being an event boundary. Once boundary scores are computed, all but the scores affected by the zero-padding are shared through multiple RTP passes.

With the computed boundary scores, we then select an index that corresponds to an event boundary. For this process, we form a categorical distribution of the boundary scores. To reduce excessive randomness, we only preserve top k% scores. The boundary frame index is then determined by sampling with the computed distribution (Figure 4

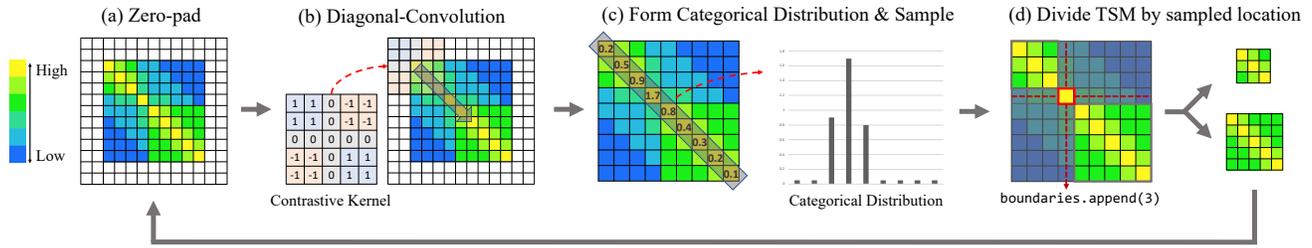


Figure 4. The figure demonstrates how Recursive TSM Parsing (RTP) works, assuming a 9x9 TSM and a 5x5 contrastive kernel. The yellow region in TSM indicates a high similarity score, while the blue region means a lower score. Using the output boundary index, TSM is divided into smaller TSMs, which again go through the same process. Note that the recurrence stops for the small sub-TSM in this example as it reached a predefined minimal length.

(d). By substituting the straightforward max operation with this sampling strategy, we can diversify training samples, compensating for the training data limitation. Now with the boundary frame index given, the TSM is *divided* into two separate TSMs, each of which is forwarded to another run of the RTP algorithm (Figure 4 (d)).

The above pass is recursively executed until one of the following end-conditions is satisfied: *a*) the parsed TSM is smaller than a predefined threshold  $T_1$ , or *b*) difference between the max boundary score and the mean boundary score is smaller than a threshold  $T_2$ . The first condition represents a prior assumption on the minimal length of an event segment, and the second condition handles long event segment cases. Note that a small difference between the highest score and the mean value implies that there are no distinguishable points.

Although it may be seen as a minor step, zero-padding plays an important role in RTP. Corner elements of the TSM are one of the following: start, end point of the video, or proximate point to the detected boundary. This indicates that corner points are unlikely to be a boundary frame. Therefore, in addition to enabling the boundary computation at corners, the zero-padding also allows for assigning relatively lower boundary scores for the corners, suppressing false or duplicated event boundary detection.

### 3.2.2 Boundary Contrastive Loss

Similar to RTP, Boundary Contrastive loss (BoCo loss) shares the same “boundary pattern” intuition in Figure 1. The objective of BoCo loss is to train the feature encoder, in order to produce informative TSM, whereas the goal of RTP is to extract boundary indices from a given TSM. With the boundary indices computed by RTP, the BoCo loss helps TSM to be more distinguishable at boundaries (Figure 3).

Inspired by the recent success of contrastive learning, BoCo loss adopts the metric learning strategy. Figure 5 explains how positive and negative samples are selected for

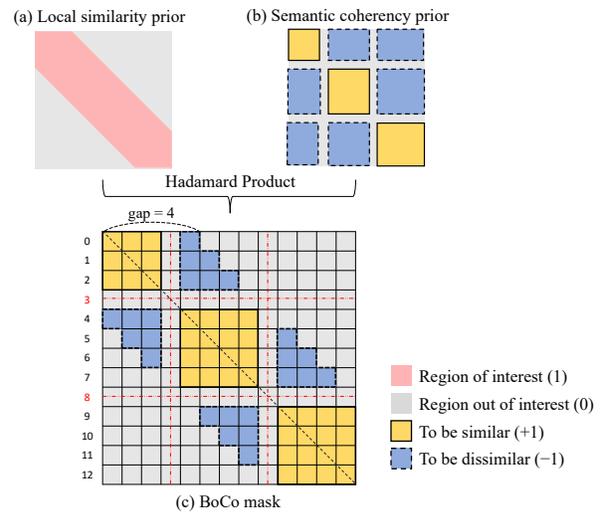


Figure 5. Given the list of boundary indices ([3,8] in the above example), (c) represents the mask to compute BoCo loss. Both prior (a), (b) are merged by element-wise multiplication.

the BoCo loss. As our task focuses on the short-term frame relationship, the similarity between distant frames would not give much information for detecting event boundaries. From this assumption, we pose *local similarity prior* as described in Figure 5 (a). The prior implies that we only care about the similarities between frames within a predefined interval, or in other words, “gap”.

With given  $n$  boundary frame indices, a video can be parsed into  $n + 1$  segments. Recall that the video frames in the same segment are *semantically coherent*, meaning that the frame similarity among them should be high, while the similarity between the frames in different segments should stay low. This assumption can be implemented as the *semantic coherency prior* mask shown in Figure 5 (b).

By using elementwise multiplication, we can combine both assumptions, getting the BoCo mask in Figure 5 (c) that represents valid positive/negative pairs. Once the posi-

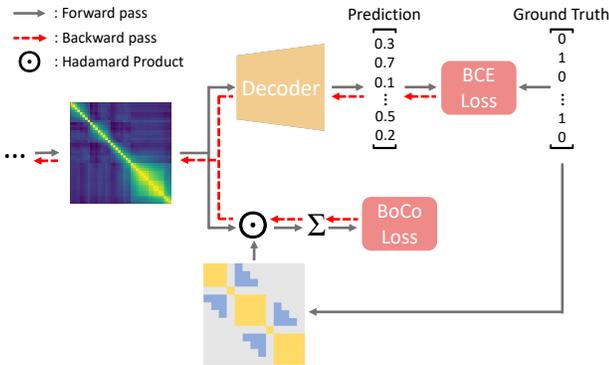


Figure 6. Supervised GEBD framework with a decoder. With supervision, we can replace RTP with a neural decoder and improve the GEBD performance with additional BCE loss.

tive/negative pairs are determined, there are several choices for the metric learning, including [23]. However, for computational efficiency and straightforward implementation, BoCo loss is simply computed as the difference between the mean values of the blue region and the yellow region in Figure 5 (c). Even though this simple approach gives satisfactory results (Section 4), the metric learning loss function for our algorithm could be a direction for our future work.

### 3.3. Supervised Boundary Contrastive Learning with Decoder

By simply replacing the pseudo-label with the ground-truth label, UBoCo can be converted into Supervised Boundary Contrastive learning (SBoCo). While this straightforward version of SBoCo works quite well, we can substitute RTP algorithm with TSM decoder (Figure 6), which consists of convolutional neural networks [19] and transformers [40]. In this approach, probabilities of being event boundary can be directly acquired from the decoder, only requiring simple post-processing (e.g. thresholding).

There are several advantages to adopting a neural TSM decoder. First, another widely used loss term, binary cross-entropy loss, can be additionally utilized. It enables the model to receive more informative gradient signals during the training procedure. Moreover, as it allows direct prediction, we can detour the recursive procedure of RTP, making faster inference possible. Lastly, thanks to the strong representation power of neural networks, we can employ multi-channel TSM, compensating for the limitation of the single-channel TSM’s expressive power. This approach shares the fundamental idea with [14], which used TSM as an interpretable intermediate representation. More details on the decoder will be provided in the supplementary materials.

	Method	F1@0.05	Average F1
Unsupervised	SceneDetect	27.5	31.9
	PA-Random	33.6	50.6
	PA	39.6	52.7
	UBoCo-Res50 (ours)	<b>70.3</b>	<b>86.7</b>
	UBoCo-TSN (ours)	<b>70.2</b>	<b>86.7</b>
Supervised	BMN	18.6	22.3
	BMN-StartEnd	49.1	64.0
	TCN-TAPOS	46.4	62.7
	TCN	58.8	68.5
	PC	62.5	81.7
	SBoCo-Res50 (ours)	<b>73.2</b>	<b>86.6</b>
	SBoCo-TSN (ours)	<b>78.7</b>	<b>89.2</b>

Table 1. Results on Kinetics-GEBD for unsupervised (top) and supervised (bottom) methods. The scores of previous methods are from [37].

## 4. Experiments

### 4.1. Benchmark Dataset

Kinetics-GEBD consists of train, validation, and test sets, each with about 18K videos from Kinetics-400 dataset [22]. Since the ground truth for the test set is not released, we use the validation set of Kinetics-GEBD as the test set as in [37] and split the train set of Kinetics-GEBD into the train (80%) and the validation (20%) for our experiments. As for the evaluation metric, we validate our experiments on F1@0.05. The value of 0.05 is the threshold of the Relation Distance (Rel.Dis.). Given a Rel.Dis., the predicted point is treated as correct if the discrepancy between the predicted and the ground truth timestamps is less than the threshold. Following the benchmark [37] and the official challenge, we analyze our results mainly on this threshold value. In Table 1, we also provide the average F1 score, which is the mean of the F1 scores using the thresholds from 0.05 to 0.5 with 0.05 interval. As we show in the supplementary materials, the results for other thresholds still show the competitiveness of our methods.

### 4.2. Implementation Details

We used ResNet-50 [19] pretrained on ImageNet [11] and the weights from torchvision [31] as our main backbone for fair comparisons with previous methods [37], which also used the same features. We additionally tested with TSN [41] pretrained on Kinetics dataset [22] to maximize the model performance. All experiments were conducted on a single Nvidia RTX 2080 Ti GPU equipped machine. We trained our models with the batch size of 32 using AdamW optimizer with the learning rate of 1e-3. Our model’s encoder consists of 1D CNNs and Mixer [38] for capturing short-term and long-term representations respectively. More details about our feature extraction and models are described in the supplementary materials.

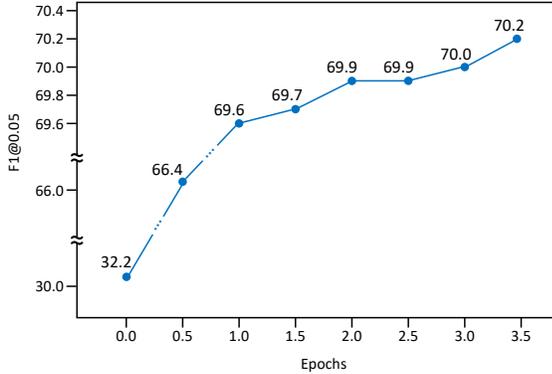


Figure 7. Improvement of UBoCo as the encoder of the model is trained on pseudo-labels in a self-supervised manner.

### 4.3. Results on Kinetics-GEBD

Table 1 illustrates the results of our models compared to prior works in unsupervised and supervised conditions. Our models outperform previous models by a large margin for not only the unsupervised setting (30.7% higher than the SOTA) but also the supervised setting (10.7% better than the SOTA) with the same extracted features (ResNet-50). We can demonstrate that our *unsupervised* model, UBoCo, is extremely powerful to surpass the previous *supervised* state-of-the-art model by a good margin of 7.8%. We notice that with the TSN features, extracted by the video-level model TSN, there is a remarkable improvement for the supervised setting, increasing the performance by 16.2% over the previous SOTA.

### 4.4. Ablation Studies

#### 4.4.1 Self-supervision with Pseudo-label

From the unsupervised model (UBoCo), we can make the pseudo-labels as explained in Section 3.2. Using these pseudo-labels, we can train our UBoCo model with gradient-descent, which can be seen as a self-supervised approach. The progressive improvement of the performance is illustrated in Figure 7. We can observe that the accuracy drastically increases as the training evolves, showing the power of self-supervision with BoCo loss. Another interesting observation is the performance at epoch 0 (33.2%), which is comparable to the existing state-of-the-art unsupervised model (39.6%). It indicates that even with under-trained feature encoder, our RTP can catch some obvious event boundaries.

#### 4.4.2 Recursive TSM Parsing

The effectiveness of the Recursive TSM Parsing (RTP) algorithm with zero-padding against other straight-forward algorithms is shown in Table 2. To extract event boundaries

	UBoCo-Res50	UBoCo-TSN
Thresholding	27.3	27.3
Local Maxima	68.1	68.8
RTP w.o. ZP	55.3	54.8
(ours) RTP	70.3	70.2

Table 2. F1@0.05 scores corresponding to different parsing algorithms. ZP stands for Zero-Padding. They share the same procedure until the boundary scores are calculated. For the method *Thresholding*, the points whose sigmoid value of event score is larger than the threshold are predicted as event boundaries. We use the best score for various thresholds from 0.1 to 0.5.

Supervision	Decoder	SBoCo-Res50	SBoCo-TSN
		70.3	70.2
✓		71.1 (+0.8)	75.5 (+5.3)
✓	✓	<b>73.2 (+2.9)</b>	<b>78.7 (+8.5)</b>

Table 3. F1@0.05 scores for the effect of supervision and decoder.

from a TSM, some other options include simple thresholding and finding local maxima of diagonal boundary scores [37]. As can be observed, RTP yields the best performance and zero-padding is essential for RTP process. We attribute the performance improvement to its coincidence with the GEGBD’s underlying assumption, “one level deeper semantics”. In other words, the recursive fashion of RTP easily catches the inherent hierarchy of action segments in that each recursion stage represents a different level of hierarchy. Figure 8 qualitatively illustrates the performance of RTP. It shows that RTP detects big changes in the early stages and starts to pick up subtle changes in an iterative way.

#### 4.4.3 Extension to Supervised Model (SBoCo)

By converting the pseudo-label in Section 3.2 into human-annotated ground truth, we can outstretch the unsupervised model (UBoCo) to the supervised model (SBoCo). Furthermore, as explained in Section 3.3, we subjoin the decoder layer so that the model can exploit the TSM not only explicitly but also implicitly. The results for both methods are shown in Table 3, demonstrating that the supervision and the decoder layer help the performance enhancement, especially for the SBoCo-TSN model. By converting input from image-level features (ResNet-50 pretrained on ImageNet) to video-level (TSN pretrained on Kinetics), we also observed additional improvement of the performance, achieving the state-of-the-art in Kinetics-GEGBD.

Moreover, to validate the role of BoCo loss in the supervised setting, ablation study about BoCo loss was conducted. Experimental result shows that accompanying BoCo loss with BCE loss can enhance the performance in

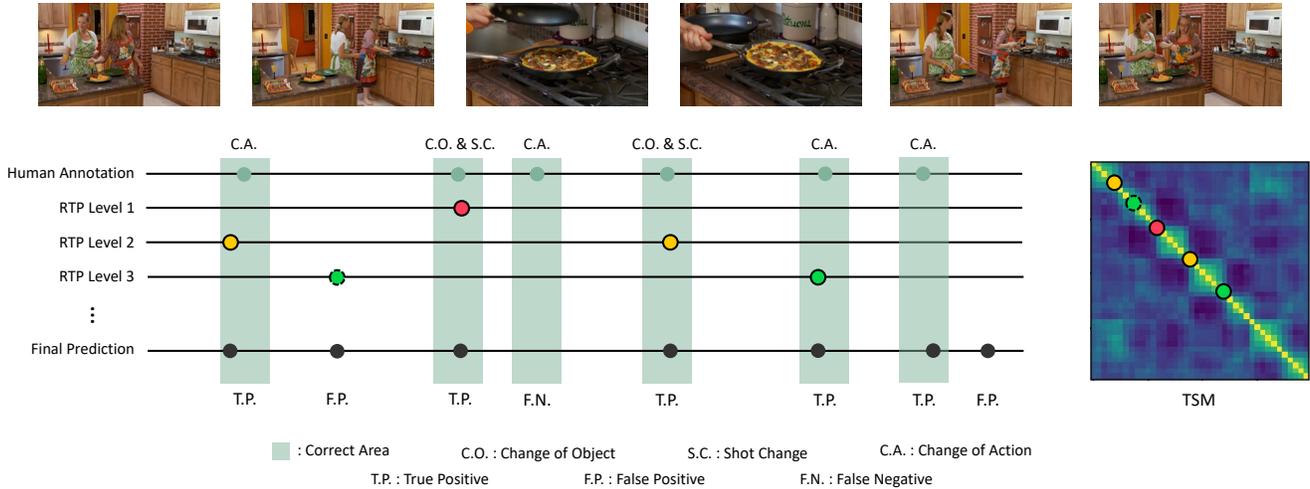


Figure 8. Above figure illustrates how RTP detects event boundaries from the given TSM. As shown in the figure, apparent boundaries including shot change are captured at the early level of RTP, while more subtle boundaries are deferred to the last level.

	SBoCo-Res50	SBoCo-TSN
BCE loss only	71.8	77.5
BCE loss & BoCo loss	<b>73.2 (+1.4)</b>	<b>78.7 (+1.2)</b>

Table 4. F1@0.05 scores without and with BoCo loss in supervised model (SBoCo).

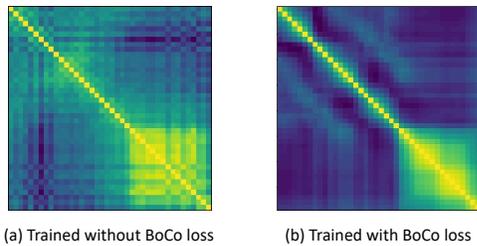


Figure 9. BoCo loss in supervised model makes the TSM more interpretable and informative. Boundary patterns in (b) is much more distinguishable than those in (a).

both ResNet feature and TSN feature (Table 4). Qualitative result (Figure 9) also illustrates that BoCo loss makes the TSM of a given video more interpretable.

## 5. Discussion and Conclusion

Generic Event Boundary Detection (GEBD) has the potential of being a fundamental upstream task for further video understanding in that it can make changes on prevailing video-division convention to a more human interpretable way. Rethinking the essence of the task, we found that what actually matters for event boundaries is *local* frame relationship, implying that TSM’s diagonal lo-

cal pattern can be a meaningful cue for boundary detection. Expanding the idea, we proposed the novel unsupervised/supervised GEBD solver that utilizes the TSM’s local distinguishing similarity pattern that emerges near the event boundaries. Both of our methods achieved state-of-the-art results in both unsupervised/supervised GEBD settings, which promotes further research on the task. Moreover, focusing on the fact that our unsupervised method can produce reasonable event boundaries without any human-annotated label, it has a great potential to be stretched to other unsupervised video understanding tasks.

There are also several limitations that stimulate future works. To begin with, we used *fixed* contrastive kernel as a realization of our intuition. Despite its exceptional performance, different variation, or even learnable kernel can be adopted for future research. Furthermore, since the only available benchmark dataset for GEBD, Kinetics-GEBD, contains videos with relatively similar duration, experiments for videos with varying duration cannot be conducted. This calls for more GEBD labels for various kinds of videos, ranging from short Tiktok clips to full movies, in the near future.

## 6. Acknowledgement

This work was partly supported by Institute of Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government(MSIT), Artificial Intelligence Innovation Hub under Grant 2021-0-02068 and Artificial Intelligence Graduate School Program under Grant 2020- 0-01361.

## References

- [1] Chiraz BenAbdelkader, Ross Cutler, Harsh Nanda, and Larry Davis. Eigengait: Motion-based recognition of people using image self-similarity. In *AVBPA*, 2001. 2, 3
- [2] Chiraz BenAbdelkader, Ross G Cutler, and Larry S Davis. Gait recognition using image self-similarity. *EURASIP*, 2004. 2, 3
- [3] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. Sst: Single-stream temporal action proposals. In *CVPR*, 2017. 1
- [4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 3
- [5] Da Chen, Xiang Wu, Jianfeng Dong, Yuan He, Hui Xue, and Feng Mao. Hierarchical sequence representation with graph network. In *ICASSP*, 2020. 3
- [6] Shixing Chen, Xiaohan Nie, David Fan, Dongqing Zhang, Vimal Bhat, and Raffay Hamid. Shot contrastive self-supervised learning for scene boundary detection. In *CVPR*, 2021. 3
- [7] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020. 3
- [8] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 3
- [9] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021. 3
- [10] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *ICCV*, 2021. 3
- [11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 6
- [12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3
- [13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3
- [14] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Counting out time: Class agnostic video repetition counting in the wild. In *CVPR*, 2020. 2, 3, 6
- [15] Victor Escorcia, Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. Daps: Deep action proposals for action understanding. In *ECCV*, 2016. 1
- [16] Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. In *ICCV*, 2017. 1
- [17] Jean-Bastien Grill, Florian Strub, Florent Althé, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Dohersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 3
- [18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 3
- [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 3, 6
- [20] Simon Jenni and Hailin Jin. Time-equivariant contrastive video representation learning. In *ICCV*, 2021. 3
- [21] 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*, 2021. 1
- [22] 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. 6
- [23] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020. 3, 6
- [24] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Ioannis Kompatsiaris. Visil: Fine-grained spatio-temporal video similarity learning. In *ICCV*, 2019. 3
- [25] Christopher A Kurby and Jeffrey M Zacks. Segmentation in the perception and memory of events. *Trends in cognitive sciences*, 2008. 1, 3
- [26] Colin Lea, Austin Reiter, René Vidal, and Gregory D Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In *ECCV*, 2016. 1, 3
- [27] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *ICCV*, 2019. 1, 3
- [28] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018. 1
- [29] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018. 3
- [30] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *BSMSP*, 1967. 4
- [31] Sébastien Marcel and Yann Rodriguez. Torchvision the machine-vision package of torch. In *ACM*, 2010. 6
- [32] Jinwoo Nam, Daechul Ahn, Dongyeop Kang, Seong Jong Ha, and Jonghyun Choi. Zero-shot natural language video localization. In *ICCV*, 2021. 3
- [33] Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *CVPR*, 2021. 3
- [34] Costas Panagiotakis, Giorgos Karvounas, and Antonis Argiros. Unsupervised detection of periodic segments in videos. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, 2018. 2, 3

- [35] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *CVPR*, 2021. 3
- [36] Jeremy R Reynolds, Jeffrey M Zacks, and Todd S Braver. A computational model of event segmentation from perceptual prediction. *Cognitive science*, 2007. 1
- [37] Mike Zheng Shou, Stan W Lei, Weiyao Wang, Deepti Ghadiyaram, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. In *ICCV*, 2021. 1, 3, 6, 7
- [38] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *arXiv preprint arXiv:2105.01601*, 2021. 6
- [39] Barbara Tversky and Jeffrey M Zacks. Event perception. *Oxford handbook of cognitive psychology*, 2013. 1
- [40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3, 6
- [41] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 6
- [42] Huiifen Xia and Yongzhao Zhan. A survey on temporal action localization. *Access*, 2020. 1
- [43] Jeffrey M Zacks, Nicole K Speer, Khena M Swallow, Todd S Braver, and Jeremy R Reynolds. Event perception: a mind-brain perspective. *Psychological bulletin*, 2007. 1