# Multi-Type Incremental Local-to-Global Context Encoder for Video Boundary Detection

Xiaoxuan Tian, Ruifan Zhao, Zhilong Ou, and Hongxing Wang
School of Big Data and Software Engineering
Chongqing University, Chongqing 401331, China
txx@stu.cqu.edu.cn, ruifanz@stu.cqu.edu.cn, zlou@stu.cqu.edu.cn, ihxwang@cqu.edu.cn

***Abstract*—Video boundary detection (VBD) aims to identify salient temporal transitions in videos, such as event and scene changes. Learning a shared model that can handle both boundary types remains challenging. To address this problem, we introduce a multi-type video boundary detection framework. The framework adopts a Local-to-Global Context Encoder as a shared backbone to jointly capture local dynamics and global contextual dependencies for both event and scene boundaries. To train the proposed framework, we design an incremental learning strategy that integrates a teacher–student paradigm, experience replay, and feature alignment to preserve prior knowledge while adapting to new data. Experimental evaluations validate that the proposed method generalizes well across event and scene boundary detection tasks, providing an effective solution for multi-type video boundary detection. Code is available at https://github.com/Xxxseventea/GLM.**

***Index Terms*—Video Boundary Detection, Multi-Type Detection Framework, Incremental Local-to-Global Context Encoder**

## I. Introduction

Video boundary detection (VBD) seeks to identify salient temporal transition points within a video stream. Accurate VBD supports downstream applications such as video summarization [1], [2] and temporal segmentation [3], [4]. Depending on the temporal scale and semantic abstraction, VBD is commonly instantiated as event boundary detection [5]–[7], or as scene boundary detection [8].

Most methods treat event and scene boundary detection as separate tasks, where dedicated models are designed and trained for individual tasks on their respective datasets. This task-specific paradigm requires maintaining multiple task-specific networks. A natural question is whether a shared model can handle both event and scene boundaries in a unified way. In this setting, we encounter two distinct challenges. First, the two tasks have different temporal requirements: event boundaries usually reflect local temporal dependencies [6], whereas scene boundaries require modeling global contextual dependencies across video sequences [9]. A shared backbone should be able to capture both local and global contextual representations. Second, existing datasets for event and scene boundary detection are collected and annotated independently, with non-overlapping label spaces and data distributions. As a result, there is no benchmark that provides both boundary types on the same videos, and a unified model must be trained on multiple heterogeneous datasets.

These observations naturally lead to a unified multi-dataset learning setting, where a single model is expected to handle both event and scene boundary detection across heterogeneous datasets. However, this unified setting is not well supported by current architectures and training paradigms. On the one hand, most VBD approaches [7], [10]–[12] rely on 3D CNNs or transformer-based encoders that are typically optimized for a single task: they either operate on local temporal windows, or become expensive and unstable on long video sequences. Temporal Perceiver [5] moves toward more general boundary detection modeling, but still relies on precomputed external features and lacks mechanisms for continual learning. On the other hand, these architectures are usually developed under a single-dataset assumption. When extended to multiple datasets via sequential training, the model suffers from catastrophic forgetting due to the lack of explicit mechanisms for incremental adaptation and cross-dataset generalization. Consequently, reusing existing backbones and training schemes does not yield a unified model across boundary types and datasets.

To address the above challenges, we propose a multi-type video boundary detection framework. The framework adopts a Local-to-Global Context Encoder as the shared backbone to jointly capture local and global contextual representations for both event-level and scene-level boundaries. We further employ a teacher–student paradigm with experience replay to support incremental adaptation to new datasets while preserving prior knowledge. In addition, a discriminator is introduced to align feature distributions and enhance cross-dataset invariance. Collectively, these components yield a shared model for event and scene boundary detection, achieving robust performance across diverse datasets.

In a nutshell, our contributions include:

- We introduce a multi-type video boundary detection framework, in which a shared model handles both event and scene boundary detection across multiple datasets.
- We design a Local-to-Global Context Encoder that simultaneously captures local and global context dependencies, unifying the representation learning of event and scene boundaries.
- We conduct experiments on benchmark datasets to validate the effectiveness of our method for multi-type video boundary detection.

Corresponding author: Hongxing Wang

## II. Related Work

### A. Video Boundary Detection

*1) Scene Boundary Detection:* Video scene boundary detection aims to divide a video into semantically coherent segments [9]. Early approaches mainly relied on unsupervised learning [13], [14], clustering adjacent shots into scenes, but their dependence on hand-crafted similarity measures limited generalization performance. In recent years, with the advancement of deep learning, both supervised [5], [8], [15]–[17] and self-supervised [18], [19] methods have significantly improved detection accuracy. Supervised methods enhance boundary identification by incorporating temporal modeling and contextual encoders, while self-supervised frameworks eliminate manual annotations through temporal contrastive learning, improving model scalability and generalization..

*2) Event Boundary Detection:* Shou et al. [6] introduced the task of generic event boundary detection, which aims to identify event boundaries consistent with human perception. Existing approaches can be broadly categorized into supervised and self-supervised methods. Supervised methods [7], [10]–[12], [20], [21] extract features using backbone networks and perform binary classification within a sliding-window framework through temporal modeling architectures. In contrast, self-supervised methods [22], [23] leverage temporal continuity or motion cues as supervision signals, applying contrastive learning frameworks with optical flow or feature variation saliency measures to achieve robust boundary detection without manual annotations.

### B. Continual Learning

Continual learning aims to enable a model to retain existing knowledge while acquiring new knowledge. Existing approaches are broadly categorized into regularization-based [24], [25], experience replay-based [26], [27], and optimization-based methods [28], [29]. Despite these advances, catastrophic forgetting, computational overhead, and limited adaptability to real-world environments remain challenges. Moreover, these methods are primarily designed for static image classification and lack explicit mechanisms to handle temporal sequence modeling in video understanding. Inspired by [30], our work addresses this gap by integrating knowledge distillation, experience replay, and data distribution alignment into a unified framework tailored for multi-type video boundary detection.

### C. State Space Models

State Space Models (SSM) [31] have recently attracted increasing interest for their efficiency in modeling long-range dependencies through state space formulations. Building upon this foundation, Mamba [32] introduces a selective scan mechanism and an optimized hardware-friendly design, achieving remarkable performance across various sequence modeling tasks. Given these advantages, Mamba has been successfully extended to multiple visual domains [33], [34]. Inspired by these works, this study investigates the applicability of Mamba to global context encoding for video boundary detection.

## III. Methodology

Given an input video unit (e.g., a frame or a shot) $v_{(p)}$ at position $p$, we extract its temporal context sequence $s_{(p)} = \{v_{(j)}\}_{j=p-c}^{p+c}$, where $2c+1 = n$ denotes the context length. Extracting a $d$-dimensional feature vector $\boldsymbol{x} \in \mathbb{R}^d$ for each $v \in s_{(p)}$, we obtain the feature sequence $\boldsymbol{X_p} = [\boldsymbol{x}_{p-c}, \ldots, \boldsymbol{x}_p, \ldots, \boldsymbol{x}_{p+c}]^\top \in \mathbb{R}^{n\times d}$. Suppose for boundary type $k$, we collect $N$ video unit training samples $\{v_i^{(k)}, y_i^{(k)}\}_{i=1}^N$, where $v_i^{(k)}$ is a video unit and $y_i^{(k)} \in \{0,1\}$ indicates whether it is a boundary of type $k$. For each $v_i^{(k)}$, we extract its context sequence $s_i^{(k)}$ and the corresponding feature sequence $\boldsymbol{X}_i^k$, yielding sets $S_k = \{s_i^{(k)}\}$ and $\mathcal{X}_k = \{\boldsymbol{X}_i^{(k)}\}$. As illustrated in Fig. 1, to handle sequentially presented training data of different boundary types, we incrementally rely on $\mathcal{X}_1$ and $\mathcal{X}_2$ to learn a Local-to-Global Context Encoder via a staged teacher–student training strategy for multi-type boundary detection. In this study, we consider scene and event boundaries as two representative boundary types.

### A. Local-to-Global Context Encoder

To effectively capture both short-term local dynamics and long-range contextual dependencies in video data, we propose to train a Local-to-Global Context Encoder:

$$\boldsymbol{Z} = f(\boldsymbol{X}; \theta), \tag{1}$$

where $\boldsymbol{X}$ is either from $\mathcal{X}_1$ or $\mathcal{X}_1$, $\boldsymbol{Z}$ is the encoder output, and $\theta$ denotes the trainable parameters. The architecture of $f$ consists of local and global encoder components.

For local context encoder, as illustrated in Fig. 2, we apply multi-head self-attention [35] with $H$ heads. For each head $h \in 1, 2, \cdots, H$, we linearly project $\boldsymbol{X}$ (after layer normalization) into query, key, and value matrices:

$$\boldsymbol{Q}_h = \boldsymbol{X}\boldsymbol{W}_h^Q, \quad \boldsymbol{K}_h = \boldsymbol{X}\boldsymbol{W}_h^K, \quad \boldsymbol{V}_h = \boldsymbol{X}\boldsymbol{W}_h^V, \tag{2}$$

where $\boldsymbol{W}_h^Q, \boldsymbol{W}_h^K, \boldsymbol{W}_h^V \in \boldsymbol{R}^{d\times d_h}$. To capture local dependencies, each head restricts its attention to a fixed-size window around each column-wise position. Let $r$ be the window radius The attention weight between positions $i$ and $j$ is computed only if $|i-j| \le r$. This is implemented by a masking matrix $\boldsymbol{B} \in \{0,1\}^{n\times n}$ with entries:

$$b_{ij} = \begin{cases} 0, & \text{if } |i-j| \le r, \\ -\infty, & \text{otherwise.} \end{cases} \tag{3}$$

The output of head $h$ is then:

$$\text{head}_h = \text{softmax}\left(\frac{\boldsymbol{Q}_h\boldsymbol{K}_h^\top + \boldsymbol{B}}{\sqrt{d_h}}\right)\boldsymbol{V}_h, \tag{4}$$

where the softmax is applied row-wise, and positions with mask value 0 receive effectively attention.

Finally, the outputs of all heads are concatenated and linearly transformed into the local context embedded representation:

$$\boldsymbol{Z}_\text{L} = \text{Concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_H)\boldsymbol{W}^O, \tag{5}$$

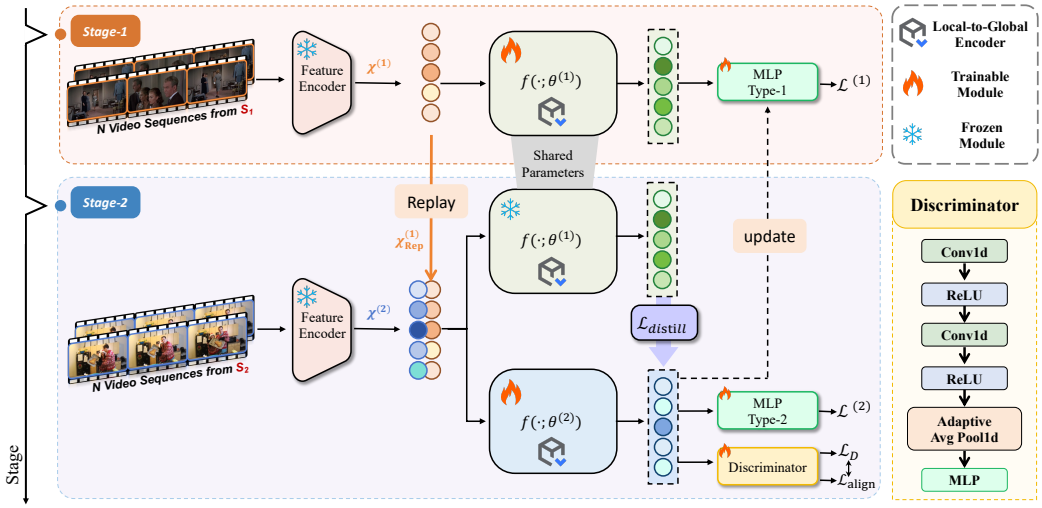


Fig. 1: The overview and detailed structures of the proposed framework, which consists of a shared Local-to-Global Context Encoder with teacher–student branches for knowledge distillation, and an data distribution alignment module with an adaptive aggregation pooling discriminator.

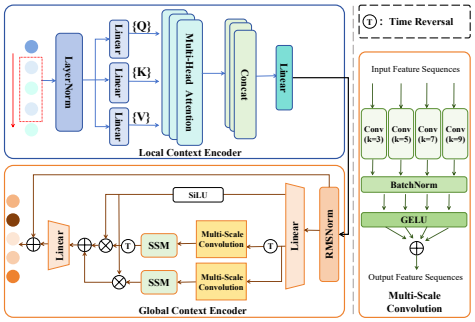


Fig. 2: Architecture of the Local-to-Global Context Encoder.

where linear projection $\boldsymbol{W}^O \in \mathbb{R}^{Hd_h \times d}$. This yields $\boldsymbol{Z}_{\mathrm{L}}$ of the same dimension $n \times d$ as the input $\boldsymbol{X}$.

For global context encoder, as illustrated in Fig. 2, we borrow the structure of Vision Mamba [36], but introduce multi-scale convolution (MSConv) and apply it bidirectionally. The MSConv is implemented with multiple groups of convolution kernels of varying sizes, followed by Batch Normalization [37] and GELU activation [38]. The organization of the remaining components of the global encoder, including RMSNorm [39], linear projection, SSM (State Space Model) [31], and SiLU activation [40], is consistent with the Vision Mamba architecture. Given the local context embedded representation $\boldsymbol{Z}_{\mathrm{L}}$, we define the global encoder process by:

$$\boldsymbol{Z} = \mathrm{GlobalContextEncoder}(\boldsymbol{Z}_{\mathrm{L}}; \theta_{\mathrm{GE}}), \tag{6}$$

with $\theta_{\mathrm{GE}}$ being the trainable parameters.

### B. Incremental Learning Strategy

*1) Stage 1:* To obtain knowledge from type-1 boundary training data, we take each video feature sequence $\boldsymbol{X}_i^{(1)} \in \mathcal{X}_1$ as input to our designed Local-to-Global Context Encoder $f$, which produces context embedded representations $\boldsymbol{Z}_i^{(1)} = f(\boldsymbol{X}_i^{(1)}; \theta^{(1)})$. Subsequently, representations $\boldsymbol{Z}_i^{(1)}$ are passed to a dedicated MLP classification head for type-1 boundary prediction (parameterized by $\theta^{(1)}_{\mathrm{pred}}$).

$$\hat{\boldsymbol{y}}_i^{(1)} = \mathrm{MLP}_{\text{Type-1}}(\boldsymbol{Z}_i^{(1)}; \theta^{(1)}_{\mathrm{pred}}). \tag{7}$$

We train $f(\cdot;\theta^{(1)})$ and $\mathrm{MLP}_{\text{Type-1}}(\cdot;\theta^{(1)}_{\mathrm{pred}})$ by minimizing the following weighted binary cross-entropy loss between every predictions $\hat{y}_i^{(1)}$ with the ground truth $y_i^{(1)}$:

$$\mathcal{L}^{(1)}(\theta^{(1)}, \theta^{(1)}_{\mathrm{pred}}) = -\mathbb{E}_{\boldsymbol{X}\sim\mathcal{X}^{(1)}}\Big[\beta y_i^{(1)} \log(\sigma(\hat{y}_i^{(1)})) + (1 - y_i^{(1)}) \log(1 - \sigma(\hat{y}_i^{(1)}))\Big], \tag{8}$$

where $\sigma(\cdot)$ is the sigmoid activation function, and $\beta > 1$ controls the contribution of positive samples to the loss, mitigating class imbalance.

*2) Stage 2:* We continually learn knowledge from type-2 boundary training data through a teacher-student paradigm with experience replay and data distribution alignment. We let $f(\cdot;\theta^{(1)})$ be the frozen teacher encoder that will guide the learning of a student encoder $f(\cdot;\theta^{(2)})$ through distillation. The input to $f(\cdot;\theta^{(2)})$ is composed of $\mathcal{X}^{(2)}$ (for learning new knowledge) and an $\alpha$ random fraction of $\mathcal{X}^{(1)}$ (for replaying old experience). Denote by $\mathcal{F}$ the combined input sequences, and by $\mathcal{X}^{(1)}_{\mathrm{Rep}}$ the replayed samples from $\mathcal{X}^{(1)}$, we have

$$\mathcal{F} = \mathcal{X}^{(2)} \cup \mathcal{X}^{(1)}_{\mathrm{Rep}} = \mathcal{X}^{(2)} \cup \mathrm{Sample}(\mathcal{X}^{(1)}, \alpha), \tag{9}$$

where $\mathrm{Sample}(\cdot,\cdot)$ denotes a random sampling operation.

During distillation, given the same input $\mathcal{F}$, the student learns to align its latent representations with those of the frozen teacher by minimizing the mean squared error:

$$\mathcal{L}_{\mathrm{distill}}(\theta^{(2)}) = \frac{1}{|\mathcal{F}|} \sum_{\boldsymbol{X}\in\mathcal{F}} \left\| f(\boldsymbol{X}; \theta^{(2)}) - f(\boldsymbol{X}; \theta^{(1)}) \right\|_2^2. \tag{10}$$

To mitigate feature distribution discrepancies across different datasets and encourage the student encoder $f(\cdot;\theta^{(2)})$ to learn distribution-aligned feature representations across $\mathcal{X}^{(2)}$ and $\mathcal{X}^{(1)}$, we introduce a binary discriminator $D(\cdot;\theta_{\mathrm{D}})$ (parameterized by $\theta_{\mathrm{D}}$) to correctly identify the source of each input $\boldsymbol{X} \in \mathcal{F}$, where those from $\mathcal{X}^{(1)}_{\text{rep}}$ are labeled 0 and those from $\mathcal{X}^{(2)}$ are labeled 1. The structure of the discriminator $D(\cdot;\theta_{\mathrm{D}})$ is depicted in Fig. 1, which sequentially applies two 1D convolution-ReLU pairs, followed by 1D adaptive average pooling. We define the adversarial objective between discriminator $D(\cdot;\theta_{\mathrm{D}})$ and encoder $f(\cdot;\theta^{(2)})$ as:

$$\begin{aligned}\mathcal{L}_{\text{adv}}(\theta_{\mathrm{D}},\theta^{(2)}) = \mathbb{E}_{\boldsymbol{X}\sim\mathcal{X}^{(2)}}\Big[\log\sigma\big(D(f(\boldsymbol{X};\theta^{(2)});\theta_{\mathrm{D}})\big)\Big] \\ + \mathbb{E}_{\boldsymbol{X}\sim\mathcal{X}^{(1)}_{\text{Rep}}}\Big[\log\big(1-\sigma\big(D(f(\boldsymbol{X};\theta^{(2)});\theta_{\mathrm{D}})\big)\big)\Big].\end{aligned} \tag{11}$$

Training discriminator $D(\cdot;\theta_{\mathrm{D}})$ needs maximizing $\mathcal{L}_{\mathrm{D}}(\theta_{\mathrm{D}}) = \mathcal{L}_{\text{adv}}$ with respect to $\theta_{\mathrm{D}}$, while training encoder $f(\cdot;\theta^{(2)})$ needs minimizing $\mathcal{L}_{\text{align}}(\theta^{(2)}) = \mathcal{L}_{\text{adv}}$ with respect to $\theta^{(2)}$.

Along with $\mathcal{L}_{\text{distill}}(\theta^{(2)})$ and $\mathcal{L}_{\text{align}}(\theta^{(2)})$, we also need a predication loss (for type-2 boundary) $\mathcal{L}^{(2)}(\theta^{(2)},\theta^{(2)}_{\text{pred}})$ to jointly train encoder $f(\cdot;\theta^{(2)})$. Simiar to (8), we define $\mathcal{L}^{(2)}(\theta^{(2)},\theta^{(2)}_{\text{pred}})$ by weighted binary cross-entropy loss between every predictions $\hat{y}^{(2)}_i$ with the ground truth $y^{(2)}_i$:

$$\begin{aligned}\mathcal{L}^{(2)}(\theta^{(2)},\theta^{(2)}_{\text{pred}}) = -\mathbb{E}_{\boldsymbol{X}\sim\mathcal{X}^{(2)}}\Big[\beta y^{(2)}_i\log(\sigma(\hat{y}^{(2)}_i)) \\ + (1-y^{(2)}_i)\log(1-\sigma(\hat{y}^{(2)}_i))\Big],\end{aligned} \tag{12}$$

where $\hat{\boldsymbol{y}}^{(2)}_i$ is predicted by a dedicated MLP classification head (parameterized by $\theta^{(2)}_{\text{pred}}$) in the following:

$$\hat{\boldsymbol{y}}^{(2)}_i = \text{MLP}_{\text{Type-2}}(f(\boldsymbol{X}^{(2)}_i;\theta^{(2)});\theta^{(2)}_{\text{pred}}). \tag{13}$$

As a result, the overall loss of $f(\cdot;\theta^{(2)})$ and $\text{MLP}_{\text{Type-1}}(\cdot;\theta^{(2)}_{\text{pred}})$ is given by

$$\mathcal{L}_{\text{overall}}(\theta^{(2)},\theta^{(2)}_{\text{pred}}) = \mathcal{L}^{(2)} + \lambda_{\text{distill}}\mathcal{L}_{\text{distill}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}}, \tag{14}$$

where $\lambda_{\text{distill}}$ and $\lambda_{\text{align}}$ are hyper-parameters for loss trade-off.

Once $\theta^{(2)}$ is trained in this stage, we update $\text{MLP}_{\text{Type-1}}(\cdot;\theta^{(1)}_{\text{pred}})$ by retraining it on the replayed data $\mathcal{X}^{(1)}_{\text{Rep}}$ using the newly learned encoder $f(\cdot;\theta^{(2)})$.

## IV. Experiment

### A. Settings

*1) Datasets:* We evaluate our method on two widely used benchmarks: **MovieNet** [41] and **Kinetics-GEBD** [6]. MovieNet [41] is a large-scale dataset for video understanding containing around 1,100 movies, among which 318 are annotated with scene boundaries, forming the **MovieScenes** subset [8] for experiments. Kinetics-GEBD [6] comprises about 54,691 videos annotated for generic event boundary detection. Its annotations focus on semantic event transitions, emphasizing temporal dynamics and fine-grained event boundaries. We evaluate the event-level performance of our method on the validation set of Kinetics-GEBD [6].

*2) Metrics:* For event boundary detection, we follow the evaluation protocol of [5], [6] and adopt the F1 score under different relative distance thresholds. For scene boundary detection, we adopt the same metrics as in previous works [8], [15]–[17], including Average Precision (AP), mean Intersection over Union (mIoU), and F1-score. These metrics assess the effectiveness of video scene detection, where higher values indicate better performance.

*3) Implementation Details:* The model takes $n = 21$ shots (for MovieNet [41]) or frames (for Kinetics-GEBD [6]) as input. In scene-level feature extraction, we capture both foreground and background information. The foreground features are extracted using a ResNet-50 [45] pre-trained on ImageNet [46], while the background features are obtained from a ResNet-50 [45] pre-trained on Places365 [47]. This design is motivated by the observation that scene boundaries are often accompanied by changes in environment or layout, and background information can influence the semantic interpretation of scene transitions. In contrast, event boundary detection primarily reflects variations in foreground motion or interactions, with relatively stable backgrounds; therefore, we only use foreground features for frame-level processing.

We set replay ratio $\alpha = 0.3$, window radius $r = 4$, positive sample weight $\beta = 5$, and loss coefficients $\lambda_{\text{distill}} = 0.5$, $\lambda_{\text{align}} = 0.1$. The Adam optimizer is employed with weight decay of $10^{-4}$, and initial learning rate of $10^{-4}$. We apply a linear warmup strategy, followed by cosine decay scheduling.

### B. Comparison with Other Continual Learning Strategies

We compare our method with continual learning strategies EWC [42], BECAME [43] and PGM [44]. For a fair comparison, we adopt only their continual learning strategies. As Table I shows, our method achieves superior results on most metrics across both training orders, verifying the effectiveness of the proposed continual learning strategy.

### C. Effectiveness of Learning Modules

We conduct ablation experiments to analyze the contributions of three key components: local encoder, global encoder, and discriminator. Specifically, we retain experience replay and feature distillation, then sequentially add local encoder, global encoder, and discriminator. As can be seen from Table II, incorporating all components leads to the best overall performance across most metrics.

### D. Versatility of Local-to-Global Context Encoder

We evaluate the performance of our Local-to-Global Context Encoder when specifically trained for different video boundary detection tasks. As reported in Table III, the evaluation is conducted on scene boundary detection and event boundary detection separately. Although the proposed Local-to-Global Context Encoder does not always achieve the best results on every individual metric, it yields competitive performance on both tasks, highlighting its strong generality and adaptability. Fig. 3 shows example results, demonstrating the versatility of the Local-to-Global Context Encoder on different video boundary detection tasks.

TABLE I: Performance comparison of incremental learning strategies under different training orders: Scene → Event and Event → Scene. The best results are in **bold** and the second-best are underlined.

| Method | Scene → Event | | | | | | Event → Scene | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Eval on Scene | | | Eval on Event | | | Eval on Event | | | Eval on Scene | | |
| | mAP | mIoU | F1 | F1@0.05 | F1@0.10 | Avg F1 (0.05∼0.50) | F1@0.05 | F1@0.10 | Avg F1 (0.05∼0.50) | mAP | mIoU | F1 |
| EWC (PNAS'17) [42] | 31.2 | 25.5 | 35.0 | 51.9 | 65.0 | 64.6 | 39.3 | 43.7 | 44.0 | 35.9 | 27.2 | 36.6 |
| BECAME (ICML'25) [43] | 49.7 | 43.0 | 45.6 | 64.4 | **82.1** | 86.3 | 64.1 | 81.9 | 86.2 | 55.6 | 48.1 | 54.1 |
| PGM (ICML'25) [44] | **55.6** | 45.5 | **54.6** | **64.6** | **82.1** | 86.5 | **65.8** | **82.5** | **86.9** | 56.0 | 49.0 | 54.2 |
| Ours | 53.4 | **47.1** | 52.6 | 64.2 | 81.9 | **86.7** | 64.7 | 82.1 | 86.5 | **57.4** | **50.2** | **55.4** |

TABLE II: Effect of the proposed modules undedifferent training orders: Scene → Event and Event → Scene. LE, GE, and D denote the Local Encoder, Global Encoder, and Discriminator, respectively.

| LE | GE | D | Scene → Event | | | | | | Event → Scene | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Eval on Scene | | | Eval on Event | | | Eval on Event | | | Eval on Scene | | |
| | | | mAP | mIoU | F1 | F1@0.05 | F1@0.10 | Avg F1 (0.05∼0.50) | F1@0.05 | F1@0.10 | Avg F1 (0.05∼0.50) | mAP | mIoU | F1 |
| ✓ | ✗ | ✗ | 19.3 | 4.8 | 24.6 | 64.4 | 82.0 | 86.4 | 64.2 | 82.1 | 86.4 | 26.8 | 22.3 | 29.5 |
| ✓ | ✓ | ✗ | 28.5 | 19.4 | 25.7 | 64.4 | 81.3 | 85.7 | 56.7 | 76.9 | 82.0 | 54.5 | 47.4 | 53.8 |
| ✓ | ✓ | ✓ | 53.4 | 47.1 | 52.6 | 64.2 | 81.9 | 86.7 | 64.7 | 82.1 | 86.5 | 57.4 | 50.2 | 55.4 |

TABLE III: Applicability of existing architectures to multi-type VBD training. Each method undergoes dataset-specific training and evaluation. The superscript ‡ indicates methods reproduced from available implementations with reasonable tuning. The best performance for each metric is highlighted in **bold**, and the second-best is underlined.

| Method | MovieNet (Scene) | | | Kinetics-GEBD (Event) | | |
|---|---|---|---|---|---|---|
| | mAP | mIoU | F1 | F1@0.05 | F1@0.10 | Avg F1 (0.05∼0.50) |
| **Method originally designed for scene boundary detection** | | | | | | |
| LGSS‡ (CVPR'20) [8] | 51.7 | 45.4 | 51.7 | 64.1 | 81.9 | 86.2 |
| CAT‡ (AAAI'23) [48] | 52.7 | 46.0 | 52.0 | **64.6** | **82.4** | **86.7** |
| Local-to-Global Context Encoder (Ours) | **58.1** | **50.7** | **56.1** | 64.5 | 82.0 | 86.6 |
| **Method originally designed for event boundary detection** | | | | | | |
| PC‡ (ICCV'21) [6] | 45.7 | 40.8 | 46.4 | 64.3 | 81.7 | **86.6** |
| Temporal Perceiver (TPAMI'23) [5] | 53.3 | **53.2** | - | **74.8** | **82.8** | 86.0 |
| Local-to-Global Context Encoder (Ours) | **58.1** | 50.7 | **56.1** | 64.5 | 82.0 | **86.6** |

Results reported under supervised learning, where ‡ indicates methods evaluated using the same feature inputs as ours.

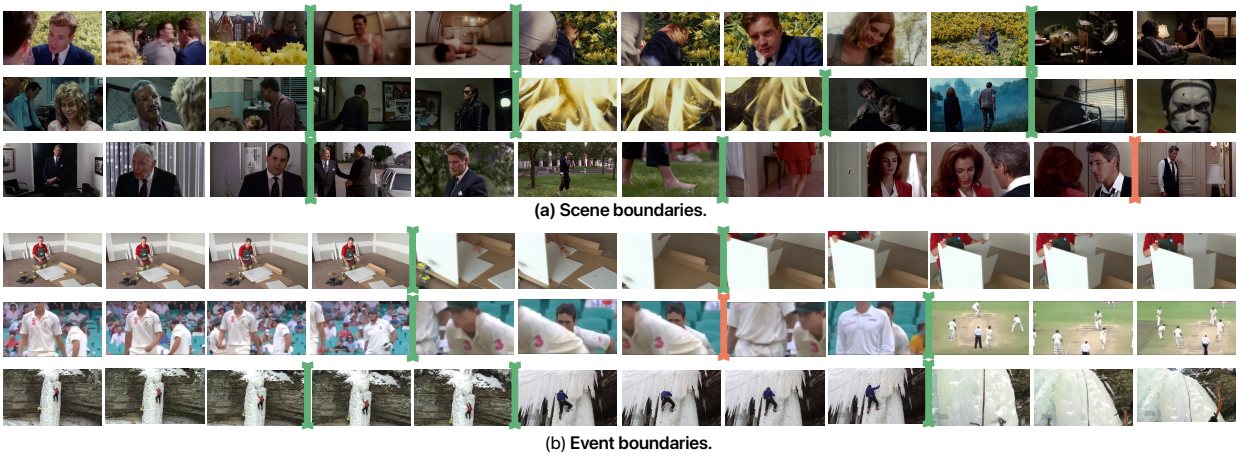


Fig. 3: Qualitative visualization of our proposed method on (a) scene boundaries in the MovieNet [41] and (b) event boundaries in the Kinetics-GEBD [6]. Correct predictions are shown in green lines, while incorrect predictions are shown in orange lines.

## V. Conclusion

In this work, we propose a multi-type video boundary detection framework that combines temporal modeling with incremental continual learning to detect both event-level and scene-level boundaries across diverse datasets. Extensive experiments demonstrate that the proposed method achieves strong generalization across event and scene boundary detection tasks. Future work will extend the model toward open-world video understanding and more efficient data adaptation.

## ACKNOWLEDGEMENTS

This work was supported in part by the Key Project of Chongqing Technology Innovation and Application Development under Grant cstc2021jscx-gksbX0033.

## References

[1] C. Pan, G. Zhang, and R. Zhong, "Trans-diff:transformer-based video summarization with diffusion," in *ICME*, 2025.

[2] J. Son, J. Park, and K. Kim, "Csta: Cnn-based spatiotemporal attention for video summarization," in *CVPR*, 2024.

[3] H. Ji, B. Chen, X. Xu, W. Ren, Z. Wang, and H. Liu, "Language-assisted skeleton action understanding for skeleton-based temporal action segmentation," in *ECCV*, 2025.

[4] C. Quattrocchi, A. Furnari, D. Di Mauro, M. V. Giuffrida, and G. M. Farinella, "Synchronization is all you need: Exocentric-to-egocentric transfer for temporal action segmentation with unlabeled synchronized video pairs," in *ECCV*, 2025.

[5] J. Tan, Y. Wang, G. Wu, and L. Wang, "Temporal perceiver: A general architecture for arbitrary boundary detection," *TPAMI*, vol. 45, no. 10, pp. 12 506 – 12 520, 2023.

[6] M. Z. Shou, S. W. Lei, W. Wang, D. Ghadiyaram, and M. Feiszli, "Generic event boundary detection: A benchmark for event segmentation," in *ICCV*, 2021.

[7] V. T. Huynh, S. Kim, H.-J. Yang, and S.-H. Kim, "Multilevel spatial–temporal feature analysis for generic event boundary detection in videos," *CVIU*, vol. 259, p. 104429, 2025.

[8] A. Rao, L. Xu, Y. Xiong, G. Xu, Q. Huang, B. Zhou, and D. Lin, "A local-to-global approach to multi-modal movie scene segmentation," in *CVPR*, 2020.

[9] E. Katz and R. D. Nolen, Eds., *The Film Encyclopedia*, 7th ed. HarperCollins, 2012.

[10] Y. Liu, L. Ma, Y. Zhang, W. Liu, and S.-F. Chang, "Multi-granularity generator for temporal action proposal," in *CVPR*, 2019.

[11] J. Li, S. Zhang, J. Wang, W. Gao, and Q. Tian, "Global-local temporal representations for video person re-identification," in *ICCV*, 2019.

[12] J. Tang, Z. Liu, C. Qian, W. Wu, and L. Wang, "Progressive attention on multi-level dense difference maps for generic event boundary detection," in *CVPR*, 2022.

[13] V. T. Chasanis, A. C. Likas, and N. P. Galatsanos, "Scene detection in videos using shot clustering and sequence alignment," *IEEE Trans. Multimedia*, no. 1, pp. 89–100, 2009.

[14] T. Lin, H.-J. Zhang, and Q.-Y. Shi, "Video scene extraction by force competition," in *ICME*, 2001.

[15] J. Tan, H. Wang, J. Li, Z. Ou, and Z. Qian, "Neighbor relations matter in video scene detection," in *CVPR*, 2024.

[16] J. Tan, H. Wang, and J. Yuan, "Characters link shots: Character attention network for movie scene segmentation," *ACM MM*, 2023.

[17] X. Wei, Z. Shi, T. Zhang, X. Yu, and L. Xiao, "Multimodal high-order relation transformer for scene boundary detection," in *ICCV*, 2023.

[18] S. Chen, X. Nie, D. Fan, D. Zhang, V. Bhat, and R. Hamid, "Shot contrastive self-supervised learning for scene boundary detection," in *CVPR*, 2021.

[19] J. Mun, M. Shin, G. Han, S. Lee, S. Ha, J. Lee, and E.-S. Kim, "Bassl: Boundary-aware self-supervised learning for video scene segmentation," in *ACCV*, 2022.

[20] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen, "BMN: Boundary-matching network for temporal action proposal generation," in *ICCV*, 2019.

[21] C. S. Lea, A. Reiter, R. Vidal, and G. Hager, "Segmental spatiotemporal cnns for fine-grained action segmentation," in *ECCV*, 2016.

[22] H. Kang, J. Kim, T. Kim, and S. J. Kim, "Uboco: Unsupervised boundary contrastive learning for generic event boundary detection," in *CVPR*, 2022.

[23] S. V. Gothe, V. Agarwal, S. Ghosh, J. R. Vachhani, P. Kashyap, and B. R. Kandur, "What's in the flow? exploiting temporal motion cues for unsupervised generic event boundary detection," in *WACV*, 2024.

[24] G. Lin, H. Chu, and H. Lai, "Towards better plasticity–stability trade-off in incremental learning: A simple linear connector," in *CVPR*, 2022.

[25] L. S. Mohammad Mahdi Derakhshani, Xiantong Zhen and C. Snoek, "Kernel continual learning," in *ICML*, 2021.

[26] L. Wang, X. Zhang, K. Yang, L. L. Yu, C. Li, L. Hong, S. Zhang, Z. Li, Y. Zhong, and J. Zhu, "Memory replay with data compression for continual learning," *ICLR*, 2022.

[27] L. Kumari, S. Wang, T. Zhou, and J. Bilmes, "Retrospective adversarial replay for continual learning," in *NeurIPS*, 2022.

[28] Y. Guo, W. Hu, D. Zhao, and B. Liu, "Adaptive orthogonal projection for batch and online continual learning," *AAAI*, 2022.

[29] S. Lin, L. Yang, D. Fan, and J. Zhang, "Beyond not-forgetting: continual learning with backward knowledge transfer," in *NeurIPS*, 2022.

[30] H. Shi and H. Wang, "A unified approach to domain incremental learning with memory: Theory and algorithm," in *NeurIPS*, 2023.

[31] A. Gu, K. Goel, and C. Ré, "Efficiently modeling long sequences with structured state spaces," in *ICLR*, 2022.

[32] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," in *COLM*, 2024.

[33] X. Jin, H. Su, K. Liu, C. Ma, W. Wu, F. HUI, and J. Yan, "Unimamba: Unified spatial-channel representation learning with group-efficient mamba for lidar-based 3d object detection," in *CVPR*, 2025.

[34] J. He, K. Fu, X. Liu, and Q. Zhao, "Samba: A unified mamba-based framework for general salient object detection," in *CVPR*, 2025.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017.

[36] L. Zhu, B. Liao, Q. Zhang, X. Wang, W. Liu, and X. Wang, "Vision mamba: efficient visual representation learning with bidirectional state space model," in *ICML*, 2024.

[37] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.

[38] D. Hendrycks and K. Gimpel, "Bridging nonlinearities and stochastic regularizers with gaussian error linear units," *arXiv*, 2016.

[39] B. Zhang and R. Sennrich, "Root mean square layer normalization," in *NeurIPS*, 2019.

[40] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Networks*, vol. 107, pp. 3–11, 2018.

[41] Q. Huang, Y. Xiong, A. Rao, J. Wang, and D. Lin, "Movienet: A holistic dataset for movie understanding," *ECCV*, 2020.

[42] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *PNAS*, no. 13, pp. 3521–3526, 2017.

[43] M. Li, Y. Lu, Q. Dai, S. Huang, Y. Ding, and H. Lu, "BECAME: Bayesian continual learning with adaptive model merging," in *ICML*, 2025.

[44] F. Wan and Y. Yang, "Probabilistic group mask guided discrete optimization for incremental learning," in *ICML*, 2025.

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[46] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *IJCV*, 2015.

[47] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *TPAMI*, vol. 40, no. 6, pp. 1452–1464, 2018.

[48] Y. Yang, Y. Huang, W. Guo, B. Xu, and D. Xia, "Towards global video scene segmentation with context-aware transformer," in *AAAI*, 2023.