*applied sciences*

MDPI

*Article*

# Audio-Visual Fusion for Semantic Event Boundary Detection in Gameplay Videos

**Soohwan Lee and Hanul Sung ***

Department of Game Design and Development, Sangmyung University, 20, Hongjimun 2-gil, Jongno-gu, Seoul 03016, Republic of Korea; 202531015@sangmyung.kr
* Correspondence: hanul.sung@smu.ac.kr

**Abstract**

Detecting gameplay-state boundaries in public gameplay videos requires inferring transitions from observable audio and visual signals without access to internal telemetry. Visual signals alone are insufficient because camera motion and visual effects can occur without state transitions, while meaningful transitions may involve subtle visual changes. Audio signals can provide complementary cues, although acoustic changes do not always correspond to gameplay-state transitions. We propose Gameplay Audio-Visual Fusion (GAVF), a framework that combines source-specific audio novelty with visual boundary scores for gameplay-state boundary detection. Gameplay audio is separated into music, sound effects (SFX), and speech. Acoustic novelty is then estimated from a self-similarity matrix for each source. Each audio signal is evaluated individually, and audio-visual fusion is assessed under the same evaluation protocol. Evaluation uses 123 clips from 10 game titles, containing 316 majority-agreed boundaries annotated by five annotators, under a leave-one-game-title-out protocol. At a relative-distance tolerance of 0.05, GAVF achieves an F1 score of 0.566, compared with 0.552 for the best-performing audio-only method and 0.304 for the visual-only baseline. These results suggest that combining audio and visual signals can provide complementary evidence for gameplay-state boundary detection.

**Keywords:** gameplay video analysis; semantic boundary detection; audio novelty; source separation; self-similarity matrix

## 1. Introduction

Public gameplay videos are no longer consumed only as entertainment content. Instead, they serve as valuable information resources. Players search them to find walkthroughs, boss battles, or story cutscenes [1]. Platforms and creators utilize them for highlight generation and video summarization [2,3], while researchers use them for player behavior analysis [4,5] and data collection for embodied agents [6,7]. In these scenarios, the useful temporal unit is rarely the entire video. Users often need to access specific gameplay moments or semantically coherent gameplay segments within long recordings.

Traditionally, segmenting gameplay videos or identifying state transitions relied on internal game telemetry, event logs, or player-action traces [4,5]. Public gameplay videos, however, do not provide direct access to such internal signals, so state transitions must be inferred from observable audio and visual cues. This inference is challenging because visible changes do not consistently align with gameplay-state transitions. A combat encounter may begin without an edit cut, a boss phase may change while the camera view remains similar, and a narrative transition may occur with only subtle visual variation. Conversely, rapid

Check for updates

Academic Editor: Andrea Prati

Received: 2 November 2026
Revised: 23 November 2026
Accepted: 25 November 2026
Published: 26 November 2026

**Copyright:** © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

camera motion, heads-up display changes, particle effects, and brief interface animations may produce strong visual changes without any meaningful change in gameplay state.

These constraints motivate the need to detect gameplay-state transitions directly from public gameplay videos. We formulate this problem as semantic boundary detection: identifying transition points that divide long recordings into semantically coherent temporal units. Such boundary points can serve as intermediate structure for downstream tasks such as browsing, chaptering, and temporally grounded dataset construction.

However, existing temporal video segmentation methods do not fully address this semantic structuring task. Shot boundary detection targets edit-level discontinuities, while Generic Event Boundary Detection (GEBD) [8] is mainly developed for general videos. These methods are therefore limited when applied directly to gameplay videos, where the observable visual structure does not reliably indicate the underlying gameplay state.

Audio signals provide another source of evidence for gameplay-state transitions. Changes in background music, sound effects, and speech may reflect shifts in combat, exploration, narrative progression, or interface interaction. However, audio cues are also noisy and incomplete: background music may continue across state changes, sound effects may occur during routine actions, and speech may not align with gameplay transitions. Together, the mismatches in visual cues and the incompleteness of audio cues motivate combining audio novelty scores with visual boundary scores for semantic boundary detection in public gameplay videos.

We propose Gameplay Audio-Visual Fusion (GAVF), a framework that combines audio novelty scores with visual boundary scores. The audio branch separates gameplay audio into music, sound effects, and speech components and computes audio novelty scores from changes in acoustic features. The visual branch uses a GEBD model to obtain frame-level visual boundary scores. These audio and visual scores are then normalized, fused, and post-processed to generate final boundary candidates. Because the framework does not require internal game telemetry, event logs, or game-specific event categories, it does not depend on title-specific internal data or event definitions.

For evaluation, we construct an annotated gameplay boundary dataset consisting of 123 public gameplay clips from 10 game titles, with 316 majority-agreed gameplay-state boundaries annotated by five annotators. We evaluate all methods under a leave-one-game-title-out protocol, in which one game title is held out for testing and the remaining titles are used for validation. At a Rel.Dis. threshold of 0.05, GAVF achieves an F1 score of 0.566, compared with 0.552 for the best-performing audio-only method and 0.304 for the visual-only baseline. These results suggest that incorporating visual cues alongside audio novelty scores can provide additional evidence for gameplay-state boundary detection in public gameplay videos.

## 2. Background and Motivation

### *2.1. Generic Event Boundary Detection*

Generic Event Boundary Detection (GEBD) aims to detect temporal points that humans naturally perceive as event or action transitions, without relying on predefined categories [8]. Recent studies have proposed various deep learning approaches for this task [9–11]. Among these methods, DDM-Net [9] provides a representative GEBD-based frame-level boundary score and has reported strong performance on standard GEBD benchmarks. DDM-Net estimates temporal visual changes by comparing frame-level differences and detecting boundary candidates from dense difference maps. Specifically, instead of relying on a single pair of adjacent frames, DDM-Net aggregates differences across multiple temporal intervals and feature levels, applying a progressive attention mechanism to combine appearance and motion cues [9].

### 2.2. Audio-Based Temporal Change Analysis

Audio signals provide temporal boundary cues because acoustic patterns often change abruptly at event transitions. Various methods have been developed to extract these cues automatically. In classical audio analysis, signals are represented using spectral features such as Mel-Frequency Cepstral Coefficients (MFCCs) [12]. MFCCs compactly describe the short-time spectral envelope of an audio signal, making them useful for tracking timbral and acoustic-state changes over time. A representative novelty-based method constructs a self-similarity matrix (SSM) from frame-level feature vectors, where high similarity indicates locally stable acoustic structure and low similarity indicates a change in acoustic context [13]. By convolving the SSM with a checkerboard kernel, the method produces an audio novelty curve whose local maxima correspond to abrupt transitions between acoustically different regions.

### 2.3. Limitations of Visual and Audio Boundary Cues in Gameplay Videos

Visual boundary detectors typically rely on visual change as the primary cue for identifying temporal boundaries. In gameplay videos, however, visual changes do not consistently align with gameplay-state transitions, and this assumption is therefore frequently violated. To characterize the gap between gameplay-state transitions and established video boundary settings, we compare gameplay boundaries with the GEBD benchmark settings [8]. Within this benchmark, Kinetics-GEBD is built on the Kinetics video dataset [14] to represent event boundaries in general videos, and the TAPOS setting uses the original TAPOS dataset [15] to represent fine-grained sub-action boundaries in Olympic sports. These settings represent boundary types centered on general-video events and sports sub-actions, whereas gameplay boundaries are defined by gameplay-state transitions. We use a Kinetics-GEBD pretrained DDM-Net model to test this transfer to gameplay videos in a zero-shot setting. Table 1 provides an initial indication of this domain gap by comparing DDM-Net's reported performance on the GEBD benchmark settings with its zero-shot performance on our gameplay dataset.

**Table 1.** Reference DDM-Net benchmark scores and zero-shot transfer to gameplay (Avg. F1).

| Dataset | Avg. F1 Score |
|---|---|
| Kinetics-GEBD | 0.873 |
| TAPOS | 0.728 |
| Gameplay (zero-shot) | 0.171 |

DDM-Net performs strongly on general-video benchmarks but drops to 0.171 when transferred directly to gameplay videos. This contrast indicates that visual boundary evidence trained on general videos can overreact to non-semantic screen dynamics, such as camera rotations and screen effects, while still missing gameplay-state transitions.

Figure 1 illustrates how visual boundary evidence and audio novelty scores can diverge from gameplay-state boundaries. The curves denote the normalized audio novelty score and DDM-Net visual boundary score, where higher values indicate stronger boundary evidence.

Figure 1a shows an exploration sequence around the sea, where movement into and out of the water produces a large visual change, even though the gameplay state remains exploration. Figure 1b shows a transition from movement toward an NPC to a dialogue sequence after a fade-out, where the audio novelty score remains low despite a gameplay-state boundary. In contrast, Figure 1c shows the end of a combat sequence, where the audio novelty score provides strong boundary evidence, even though the visual change is relatively subtle. These cases show that visual boundary scores and audio novelty scores

provide complementary but incomplete boundary evidence. This motivates audio-visual fusion for semantic boundary detection, which jointly evaluates visual boundary scores and audio novelty scores rather than relying on either cue alone.

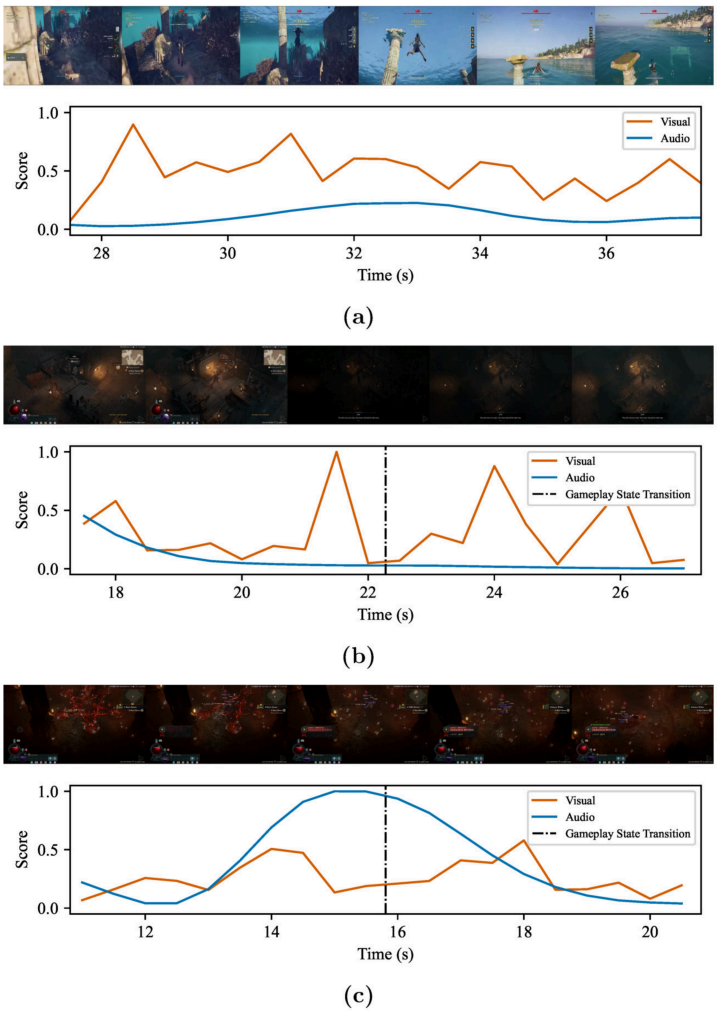



**Figure 1.** Representative mismatch cases for semantic boundary detection in gameplay videos: (**a**) a large visual change without a gameplay-state boundary, (**b**) a gameplay-state boundary with a low audio novelty score, and (**c**) a gameplay-state boundary with subtle visual change and a strong audio novelty score. Each panel shows frames around the selected time point and the corresponding audio novelty scores and visual boundary scores. The vertical dash-dotted lines in panels (**b**,**c**) mark gameplay-state transitions.

## 3. Gameplay Boundary Detection Dataset

The reliable temporal segmentation of public gameplay videos requires ground-truth (GT) boundaries corresponding to sustained changes in gameplay state or interaction context. Kinetics-GEBD and TAPOS provide closely related boundary annotations, but their annotation targets differ from the gameplay-state criterion used here, as summarized in Table 2. We therefore construct a gameplay boundary dataset with an explicit state-transition annotation policy, majority-agreed timestamps, and leave-one-game-title-out evaluation. This section describes the boundary definition, video collection, annotation protocol, consensus construction, and dataset statistics.

Kinetics-GEBD captures perceived event changes, including action, subject, environment, and shot changes, at a granularity one level below the video- or shot-level event [8]. TAPOS instead marks the internal stages of complete sports actions within a single shot [15].

Directly adopting the public datasets' annotations would evaluate their original boundary criteria. Evaluating our target requires gameplay recordings annotated under the gameplay-state policy.

**Table 2.** Comparison of boundary targets and video domains. Public-dataset properties describe their original definitions, independently of the local subsets used for transfer evaluation.

| Aspect | Kinetics-GEBD [8] | TAPOS [15] | Gameplay (Ours) |
|---|---|---|---|
| Video source | Kinetics-400 human-activity videos | Public Olympic sports videos | Public gameplay recordings without commentary |
| Boundary target | Human-perceived generic event changes | Sub-action transitions within a sports action | Sustained gameplay-state or interaction-context changes |
| Annotation unit | Kinetics video clip | One complete action instance within a single shot | Gameplay clip; 25–120 s |
| Domain coverage | Kinetics-400 activity distribution | 21 sports-action classes | 10 titles; primarily action, adventure, and role-playing gameplay |

### *3.1. Boundary Definition*

In this work, a gameplay-state boundary is defined as a temporal point at which the primary gameplay state or interaction context changes, rather than a point defined solely by visual or acoustic change. Such boundaries include transitions in objective, required player behavior, menu state, narrative context, or combat phase. They exclude micro-events that occur within the same sustained gameplay context, such as individual attacks, item pickups, or brief interface checks.

For each public gameplay clip $V$ with duration $T$, a boundary detector outputs a predicted boundary set $\hat{B} = \{\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_m\}$, where $\hat{b}_i \in [0, T]$. The corresponding ground-truth set $B$ is $B = \{b_1, b_2, \ldots, b_n\}$, where $b_j \in [0, T]$. Each $b_j$ denotes a consensus gameplay-state transition time obtained from the annotation protocol and consensus construction described in Sections 3.3 and 3.4.

### *3.2. Video Collection*

We selected gameplay clips without player commentary from the ten titles listed in Table 3.

**Table 3.** Statistics of gameplay videos used for dataset construction.

| No. | Game Title | Release Year | Clips | Duration (s) | GT Boundaries |
|---|---|---|---|---|---|
| 1 | Black Myth: Wukong | 2024 | 12 | 686 | 23 |
| 2 | Elden Ring | 2022 | 12 | 576 | 23 |
| 3 | Lies of P | 2023 | 10 | 512 | 18 |
| 4 | Diablo IV | 2023 | 11 | 689 | 37 |
| 5 | Crimson Desert | 2026 | 10 | 567 | 30 |
| 6 | Genshin Impact | 2020 | 14 | 859 | 43 |
| 7 | Monster Hunter Rise | 2021 | 16 | 1123 | 35 |
| 8 | Assassin's Creed Odyssey | 2018 | 12 | 785 | 29 |
| 9 | Horizon Forbidden West | 2022 | 9 | 635 | 35 |
| 10 | NieR: Automata | 2017 | 17 | 1035 | 43 |
| Total | | - | 123 | 7467 | 316 |

The evaluation set contains 123 clips totaling 7467 s, with a mean duration of 60.7 s and a range of 25–120 s. It was obtained from 126 selected clips by excluding three clips with no accepted consensus boundary. Detection on clips containing no consensus boundary is therefore outside the current evaluation. The dataset includes ten game titles, primarily covering action, adventure, and role-playing gameplay.

### 3.3. Annotation Protocol

Each clip was independently annotated by five annotators. Before annotation, annotators were briefed on the annotation guidelines and reviewed example cases to align their interpretation of gameplay-state boundaries. The annotation guidelines defined a boundary as a point at which the state of the game world changes meaningfully. Annotators marked a point as a boundary only when it satisfied the following three criteria:

1. Actual change in game state. The change must alter the main gameplay context, such as the player's current objective, required action, or primary focus of attention, rather than reflecting an isolated player input.
2. Exclusion of micro-events within the same state. Small or repetitive events that do not alter the current gameplay state are not considered boundaries. Examples include individual attacks during combat and item collection during exploration.
3. Formation of a new primary gameplay context. The change must establish a sustained gameplay context distinct from the preceding segment. Transitions from exploration to combat, from combat to a cutscene, or from active play to inventory, map, or equipment management can therefore constitute boundaries. Brief interface checks followed by an immediate return to the same gameplay flow are excluded.

Points that did not satisfy all three criteria were excluded. Annotators then assigned a category to the segment following each boundary. These categories are used for annotation control and dataset analysis. The evaluation targets boundary times, not segment categories.

Five reference categories were provided to maintain annotation consistency. Exploration covers movement and environmental exploration. Combat covers regular encounters and boss battles. Narrative covers cutscenes and story progression. Menu/UI covers interactions with inventories, maps, and other interfaces. Other covers boundaries that do not fit these categories.

### 3.4. Consensus Ground-Truth Construction

Candidate clusters were formed from annotations within 1.5 s before or after a seed annotation. A cluster was accepted as a GT boundary when it contained annotations from at least three distinct annotators. The boundary time was the median of all annotation timestamps in the accepted cluster, and its category was the most frequent category among those annotations. The radius limits the distance from the seed, rather than the span between every pair of annotations. If an annotator contributes multiple marks, agreement counts that annotator once, while all marks contribute to the median and category vote. Figure 2 shows accepted cluster counts and the proportion with a unique most frequent category across seed radii. The figure labels clusters with a unique most frequent category as Category-pure and clusters with a tie as Category-mixed. At a 1.5 s radius, all 316 accepted clusters have a unique most frequent category.

### 3.5. Dataset Statistics

The resulting dataset contains 123 clips and 316 GT boundaries. Exploration accounts for 34.5% of the boundaries, while Combat, Menu/UI, and Narrative are more evenly

distributed, as shown in Table 4. Among all GT boundaries, 73.7% were identified by at least four annotators, as shown in Table 5.

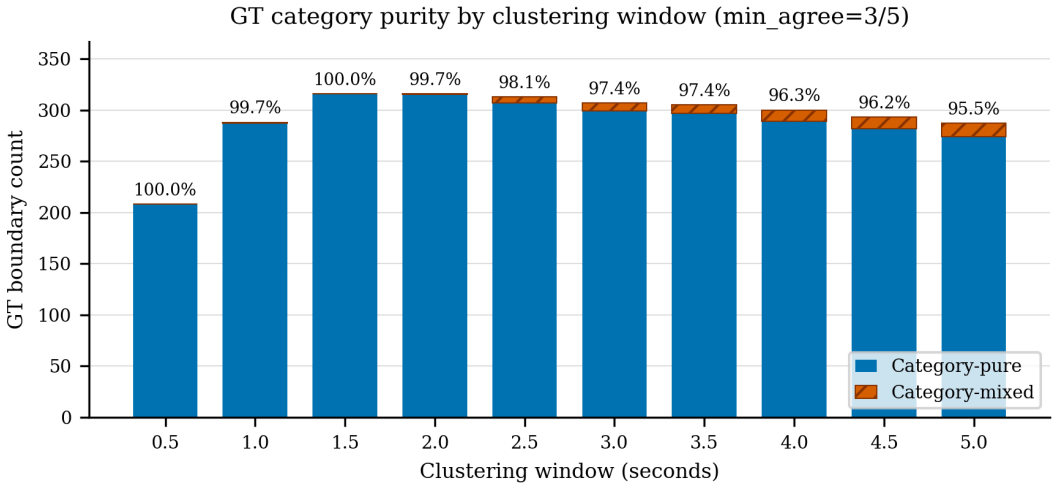



**Figure 2.** Accepted GT boundary cluster counts and the proportion with a unique most frequent category across clustering seed radii.

**Table 4.** Category distribution of GT boundaries.

| Category | Count | Ratio |
|---|---|---|
| Exploration | 109 | 34.5% |
| Menu/UI | 74 | 23.4% |
| Combat | 72 | 22.8% |
| Narrative | 61 | 19.3% |
| Total | 316 | 100.0% |

**Table 5.** Annotator agreement distribution of GT boundaries.

| Agreement Level | Count | Ratio |
|---|---|---|
| 3/5 | 83 | 26.3% |
| 4/5 | 94 | 29.7% |
| 5/5 | 139 | 44.0% |
| Total | 316 | 100.0% |

## 4. Gameplay Audio-Visual Fusion (GAVF)

GAVF is a score-level fusion framework for semantic boundary detection in gameplay videos. It computes source-separated audio novelty scores and a visual boundary score, then normalizes, aligns, and fuses them before boundary candidate generation.

### *4.1. Overview*

Figure 3 presents the overall pipeline of our framework. The input gameplay clip is processed through separate visual and audio branches. The visual branch estimates frame-level visual boundary scores using a GEBD model. The audio branch separates the mixture audio into music, SFX, and speech components and computes a novelty score for each signal. The visual and audio scores are normalized and combined using a weighted sum. Semantic boundary candidates are then generated using adaptive thresholding and a minimum-distance constraint.

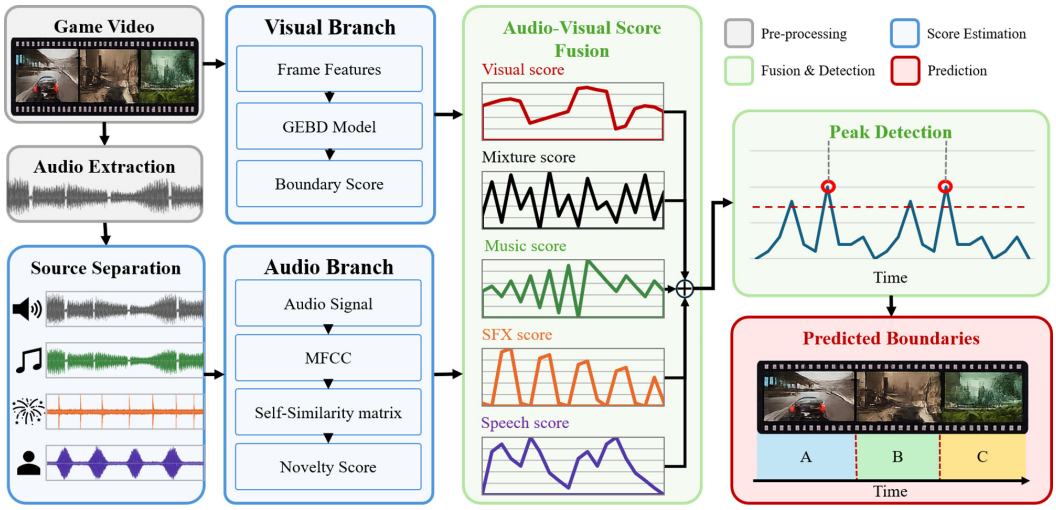



**Figure 3.** Overview of the GAVF framework for gameplay boundary detection.

### *4.2. Visual Boundary Score Extraction*

In the visual branch, we use DDM-Net as the visual boundary score extractor. DDM-Net is selected because it is an open-source GEBD model with strong reported performance on standard GEBD benchmarks and provides dense frame-level boundary scores. We use the Kinetics-GEBD pretrained weights without additional gameplay training for both the visual-only baseline and the visual branch of GAVF. Keeping the visual score extractor fixed allows performance differences to reflect score-level audio-visual fusion rather than gameplay-specific visual retraining. The resulting frame-level output $S_{\mathrm{DDM}}(t) \in [0,1]$ is used as the visual boundary score.

### *4.3. Source-Separated Audio Novelty Score Extraction*

The audio branch first separates gameplay audio into source signals, then converts each signal into frame-level acoustic features, measures temporal self-similarity, and applies a checkerboard novelty kernel to identify time points where the acoustic context changes abruptly.

#### 4.3.1. Audio Separation

Gameplay audio contains multiple signal types, including sound effects, background music, and speech. We use the pretrained Multi-Resolution CrossNet (MRX) model proposed for the Cocktail Fork problem [16] to separate the original audio mixture into music, SFX, and speech. This separation allows boundary evidence from transient in-game interactions, background music changes, and speech events to be analyzed separately rather than being mixed into a single audio novelty signal. SFX primarily contains short, impulsive sounds associated with in-game interactions, including hits and explosions. Music captures background music that typically changes over longer time scales, while speech primarily contains dialogue from gameplay and cutscenes.

#### 4.3.2. MFCC Feature Extraction

MFCC features are used to represent short-time spectral structure in a compact form before measuring temporal changes in each separated audio signal. For each audio signal, we extract Mel-Frequency Cepstral Coefficients (MFCCs) at a fixed hop size. The MFCC features are concatenated with their first- and second-order temporal differences to capture temporal variation. The resulting feature vectors are averaged over successive time windows before SSM construction. The aggregated vectors form a feature matrix $\mathbf{F} \in \mathbb{R}^{T_f \times D}$, where $T_f$ is the number of aggregated vectors and $D$ is the feature dimension.

#### 4.3.3. Self-Similarity Matrix Construction

The SSM compares aggregated acoustic feature vectors across time to represent stable contexts and transition points. A Self-Similarity Matrix (SSM) $\mathbf{S} \in \mathbb{R}^{T_f \times T_f}$ is constructed from the feature matrix $\mathbf{F}$. Each SSM element represents the cosine similarity between two aggregated feature vectors as defined in Equation (1).

$$S[i,j] = \frac{\mathbf{f}_i \cdot \mathbf{f}_j}{\|\mathbf{f}_i\| \, \|\mathbf{f}_j\|} \tag{1}$$

The SSM represents segments with maintained acoustic patterns as high-similarity blocks, while boundaries where patterns change appear as low-similarity transitions.

#### 4.3.4. Checkerboard Novelty Calculation

Checkerboard novelty calculation converts local SSM structure into a one-dimensional novelty score by contrasting within-context similarity against cross-context similarity around each time point. To emphasize temporal change points in the SSM, a checkerboard kernel is applied. Let $p$ and $q$ be discrete index offsets around index $t$. The kernel $C$ is defined in Equation (2).

$$C[p,q] = \begin{cases} +1, & (p<0 \wedge q<0) \text{ or } (p \geq 0 \wedge q \geq 0), \\ -1, & \text{otherwise}, \end{cases} \quad p,q \in \{-L, \ldots, L-1\}. \tag{2}$$

The regions $(p<0, q<0)$ and $(p \geq 0, q \geq 0)$ represent within-segment similarity before and after the boundary, respectively. The regions $(p<0, q \geq 0)$ and $(p \geq 0, q<0)$ represent cross-boundary similarity. The checkerboard kernel therefore assigns positive weights to within-segment similarity and negative weights to cross-boundary similarity.

The raw novelty score $\mathcal{N}_{\text{raw}}(t)$ at index $t$ is calculated as the local correlation between the SSM $S$ and the kernel $C$, as given by Equation (3).

$$\mathcal{N}_{\text{raw}}(t) = \sum_{p=-L}^{L-1} \sum_{q=-L}^{L-1} C[p,q] \cdot S[t+p, \; t+q] \tag{3}$$

Here, $t$ denotes a discrete SSM index and $L$ is the kernel half-width. The kernel separates indices before $t$ from indices at or after $t$. $\mathcal{N}_{\text{raw}}(t)$ becomes large when the similarity within each side of the boundary is high and the similarity between the two sides is low. Negative values are clipped to zero, and the resulting scores are normalized by their maximum within each clip to obtain $\mathcal{N}(t)$. If the maximum is zero, all novelty scores remain zero. Thus, the novelty score increases at points where acoustic characteristics change and serves as the audio evidence used for later boundary candidate generation.

#### 4.3.5. Source-Level Audio Score Aggregation

After computing a novelty curve for each separated source signal, we aggregate source-level boundary evidence using a weighted sum. Each non-negative source-level novelty curve is normalized by its maximum within the clip before aggregation. Let the normalized novelty curve be $\tilde{\mathcal{N}}_s(t)$. The audio novelty score is calculated using Equation (4).

$$\mathcal{N}_{\text{audio}}(t) = \sum_{s \in \mathcal{S}_a} w_s \cdot \tilde{\mathcal{N}}_s(t) \tag{4}$$

where $\mathcal{S}_a$ is the set of selected audio signals and $w_s$ is the weight of each signal. These scalar coefficients are selected jointly with the other fusion and peak detection parameters on the validation recordings in each LOGTO fold, as described in Section 5.2.

### 4.4. Audio-Visual Score Fusion

The visual boundary score is interpolated onto the temporal grid of the audio novelty curves and normalized by its maximum within each clip. The normalized visual boundary score is denoted as $\tilde{S}_{\mathrm{DDM}}(t)$. The aggregated audio score is used directly without additional normalization before audio-visual fusion.

Audio and visual scores capture different boundary cues, so weak evidence from one cue can be complemented by stronger evidence from the other. The aggregated audio score $\mathcal{N}_{\mathrm{audio}}(t)$ and the normalized visual score $\tilde{S}_{\mathrm{DDM}}(t)$ are combined by the weighted sum in Equation (5).

$$\mathcal{N}_{\mathrm{av}}(t) = \mathcal{N}_{\mathrm{audio}}(t) + w_{\mathrm{DDM}} \cdot \tilde{S}_{\mathrm{DDM}}(t) \tag{5}$$

where $w_{\mathrm{DDM}}$ is the weight of the visual boundary score. This scalar fusion coefficient is distinct from the frozen Kinetics-GEBD pretrained network parameters and is selected jointly with the audio coefficients and peak detection parameters on the validation recordings of each fold.

### 4.5. Boundary Candidate Generation

To extract boundary candidates from the fused audio-visual score curve $\mathcal{N}_{\mathrm{av}}(t)$, we apply peak detection. A local maximum is selected as a boundary candidate if it is above the threshold $\theta$ and is at least a minimum distance $d$, measured in score samples, away from other selected peaks. Instead of using a fixed threshold for all clips, we adopt adaptive thresholding based on the statistical distribution of score values within each clip. The sensitivity is controlled by the parameter $k$, as shown in Equation (6).

$$\theta = \mu_{\mathcal{N}} + k \cdot \sigma_{\mathcal{N}} \tag{6}$$

where $\mu_{\mathcal{N}}$ and $\sigma_{\mathcal{N}}$ are the mean and standard deviation of $\mathcal{N}_{\mathrm{av}}(t)$ within the clip, respectively. Both statistics are computed from all samples of the fused score curve in the clip. The standard deviation is the population standard deviation, with the number of samples as its denominator. The resulting threshold is constant within each clip and adapts across clips according to their score distributions and the validation-selected sensitivity coefficient. The minimum distance controls peak separation separately and does not affect these statistics. Since all clip samples are used, this procedure operates offline without using test annotations to compute the threshold. The minimum distance $d$ is used as a temporal suppression interval to prevent adjacent peaks from redundantly detecting the same boundary. The parameters $k$ and $d$ are selected on the validation set in each LOGTO fold, as described in Section 5.2.

## 5. Experiments

### 5.1. Evaluation Protocol

For evaluation, we use the GT boundary set constructed in Section 3.4. To evaluate boundary detection accuracy, we measure the temporal alignment between predicted and GT boundaries. The Generic Event Boundary Detection (GEBD) benchmark evaluates predictions using Relative Distance (Rel.Dis.) [8]. Rel.Dis. normalizes the temporal error between the predicted boundary and the GT boundary by the duration of the evaluation instance. Since we treat each gameplay clip as an evaluation instance, the Rel.Dis. between a predicted boundary $\hat{b}_i$ and a GT boundary $b_j$ is calculated using Equation (7).

$$\text{Rel.Dis.}(\hat{b}_i, b_j) = \frac{|\hat{b}_i - b_j|}{T} \tag{7}$$

where $T$ is the total length of the gameplay clip. A smaller Rel.Dis. value indicates a closer prediction, and a smaller tolerance threshold $\rho$ results in a stricter matching criterion.

At each tolerance threshold $\rho$, a predicted boundary is matched to a GT boundary if its Rel.Dis. is less than or equal to $\rho$. A matched prediction is counted as a true positive (TP), an unmatched prediction as a false positive (FP), and an unmatched GT boundary as a false negative (FN). These counts are used to calculate the standard precision, recall, and F1 scores.

For each threshold $\rho \in \{0.05, 0.10, \ldots, 0.50\}$, F1 is computed separately for each held-out game-title fold using its TP, FP, and FN counts. The fold-specific F1 scores are then averaged equally across the ten folds. This averaging gives each game title equal weight, rather than allowing titles with more boundaries and predictions to dominate the aggregate score. Following the GEBD benchmark, we retain results at all ten relative-distance thresholds. F1 at $\rho = 0.05$ is used as the primary reported metric to emphasize stricter temporal matching, while the average across the ten thresholds is reported as a secondary metric. Since the tolerance is relative to clip duration, even $\rho = 0.05$ permits larger absolute timing errors in longer clips.

### 5.2. Leave-One-Game-Title-Out Setup

Gameplay videos from the same title often share visual style, interface design, camera behavior, and sound design. Splitting the validation and test sets at the clip level can place clips from the same game title in both sets, which may overestimate performance because title-specific visual and audio cues are shared across the split. To evaluate performance on unseen game titles within the gameplay dataset, we use a cross-validation-based protocol. Specifically, each game title is treated as one group, and leave-one-group-out cross-validation is applied. In each fold, all clips from one game title are used as the test set, and clips from the remaining game titles are used as the validation set to select fusion weights and post-processing parameters. Hereafter, this setting is referred to as leave-one-game-title-out (LOGTO). This procedure is repeated for all game titles, and performance at each tolerance threshold is reported as the mean of the fold-specific F1 scores.

The experiments use the dataset constructed in Section 3, which contains 123 clips and 316 GT boundaries. All audio-based methods use the same feature extraction parameters to ensure a controlled comparison under identical feature extraction conditions. Audio signals are sampled at 22,050 Hz and represented using 13 MFCCs together with their first- and second-order temporal differences, resulting in a 39-dimensional feature vector. Before constructing the SSM, the MFCC-based feature vectors are averaged over a 3.0 s window using a nominal hop size of 0.5 s. The checkerboard kernel half-width is fixed at $L = 10$ for all clips, yielding a $20 \times 20$ kernel. At the nominal hop size, this compares approximately 5 s of context on each side of a candidate boundary, spanning approximately 10 s in total. The visual boundary score is obtained using the Kinetics-GEBD-pretrained DDM-Net model described in Section 4.2, without gameplay-specific retraining. The same visual boundary score is used for the DDM-Net baseline and for the visual branch of GAVF, so differences between these methods reflect post-processing calibration and score-level fusion rather than changes in the visual model.

Parameters are selected by exhaustive grid search over the candidate values listed in Table A3 in Appendix A. For single audio signal methods and DDM-Net, $k$ and $d$ are optimized jointly. For Audio Signal Fusion, the audio fusion weights and the post-processing parameters $k$ and $d$ are optimized jointly; for GAVF, $w_{\mathrm{DDM}}$ is additionally included in the joint search. Fusion weights are not constrained to sum to one.

In each LOGTO fold, validation comprises all clips from the remaining nine game titles. At each relative-distance threshold, validation macro-F1 is calculated as the unweighted

mean of clip-specific F1 scores. The selection objective is the average of these validation macro-F1 scores across the ten thresholds from 0.05 to 0.50, rather than F1 at 0.05 alone. Thus, clips, rather than game titles, receive equal weight during parameter selection; final test performance is averaged equally across held-out game-title folds, as described in Section 5.1. Ties are resolved using micro-F1 averaged across the same ten thresholds, with micro-F1 computed from pooled validation TP, FP, and FN counts at each threshold. One parameter configuration is selected for each method and fold and applied unchanged to all clips of the held-out game title at every evaluation threshold. Test data are not used for parameter selection. The selected configurations are reported in Appendix A.

### *5.3. Baselines*

The comparison includes audio-only, visual-only, and audio-visual methods. Each method computes a temporal boundary score in a different way. A post-processing procedure is then applied to the resulting score curve to generate final boundary candidates:

1. Single Audio Signal. Uses the audio novelty score from one of the four audio signals: Mixture, Music, SFX, or Speech.
2. Audio Signal Fusion. Normalizes and combines the audio novelty scores from multiple audio signals using a weighted sum, as defined in Equation (4).
3. DDM-Net. Uses the frame-level visual boundary score from DDM-Net pretrained on Kinetics-GEBD as the visual baseline.
4. Gameplay Audio-Visual Fusion (GAVF). Aligns and combines the fused audio novelty score with the visual boundary score using a weighted sum, as defined in Equation (5).

The adaptive thresholding and minimum-distance constraint described in Section 4.5 are applied to all score-based methods, including the audio-based methods, DDM-Net, and GAVF; the selected parameter values may differ across methods and folds. The post-processing parameters are selected on the validation set of each fold, and the test set is not used for hyperparameter selection. For Audio Signal Fusion and GAVF, the fusion weights are selected using the same validation protocol. The selected post-processing parameters and fusion weights for each held-out game-title fold are reported in Appendix A (Tables A1 and A2).

### *5.4. Results*

#### 5.4.1. Single Audio Signal Analysis

The single-audio-signal experiment evaluates the contribution of each audio signal to gameplay video boundary detection. Table 6 reports the LOGTO results for the four audio signals.

**Table 6.** Single audio signal results (LOGTO)

| | Rel.Dis. Threshold | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Signal** | **0.05** | **0.10** | **0.15** | **0.20** | **0.25** | **0.30** | **0.35** | **0.40** | **0.45** | **0.50** | **Avg** |
| Mixture | 0.508 | 0.631 | 0.680 | 0.696 | 0.708 | 0.723 | 0.729 | 0.732 | 0.741 | 0.745 | 0.689 |
| Music | 0.395 | 0.536 | 0.603 | 0.628 | 0.668 | 0.690 | 0.698 | 0.704 | 0.718 | 0.727 | 0.637 |
| **SFX** | **0.541** | **0.640** | **0.681** | **0.703** | **0.721** | **0.731** | **0.746** | **0.753** | **0.755** | **0.755** | **0.703** |
| Speech | 0.263 | 0.385 | 0.468 | 0.529 | 0.569 | 0.594 | 0.632 | 0.655 | 0.677 | 0.698 | 0.547 |

Bold scores indicate the highest value in each score column.

SFX achieves the highest performance among the single audio signals, with an Avg. F1 of 0.703. Mixture is second with 0.689, followed by music with 0.637 and speech with 0.547. SFX achieves the highest F1 score across all Rel.Dis. thresholds, indicating that it provides the strongest boundary evidence among the evaluated audio signals. This result is consistent with the role of SFX in gameplay videos, where actions, impacts, combat events, and interface interactions often generate short acoustic changes near gameplay-state

transitions. The separated SFX signal is therefore more effective for boundary detection than the original mixture in this setting.

To analyze complementarity among the audio signals, we measure the overlap between the TP sets obtained under the adaptive-threshold LOGTO setting. The Jaccard similarity between two TP sets $A$ and $B$ is defined as $|A \cap B|/|A \cup B|$. Lower values indicate that the two signals detect more distinct sets of boundaries. Figures 4 and 5 present the Jaccard similarity and shared TP count at a Rel.Dis. threshold of 0.05.

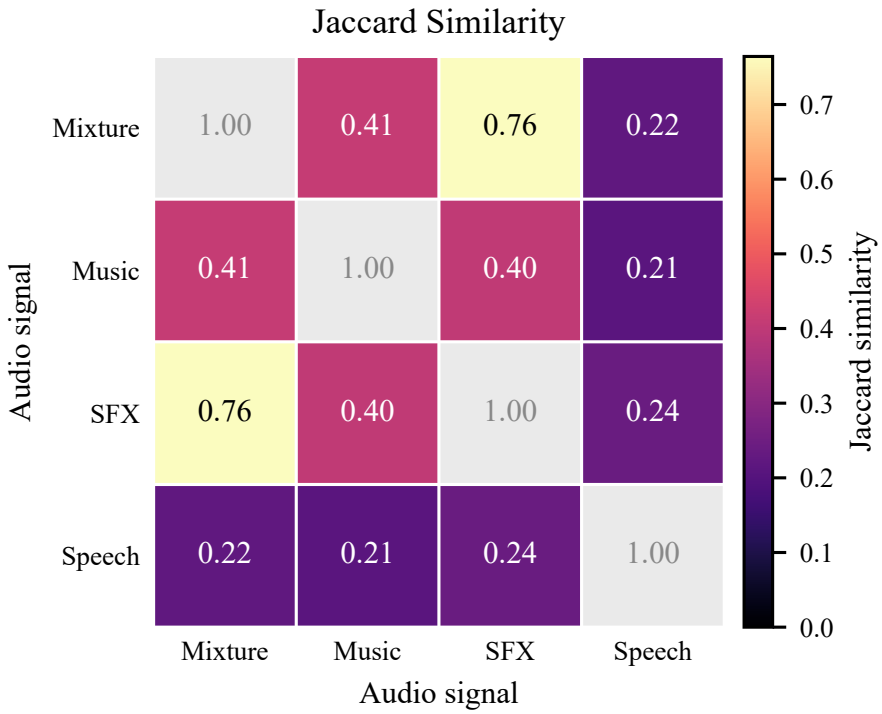



**Figure 4.** Jaccard similarity between TP sets of each signal under LOGTO at Rel.Dis. 0.05.

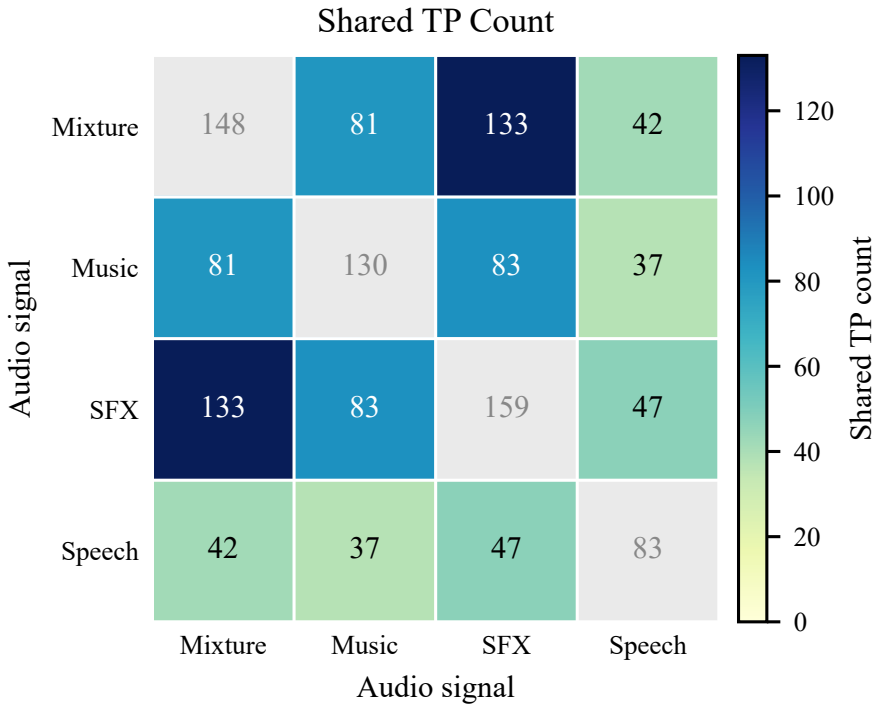



**Figure 5.** Shared TP count between audio signals under LOGTO at Rel.Dis. 0.05.

Mixture and SFX show the highest Jaccard similarity, 0.76, and the largest shared TP count, 133. Mixture and SFX therefore detect substantially overlapping sets of boundaries. Although music and speech achieve lower single-audio-signal F1 scores, their relatively low overlap with the other signals indicates that they detect different subsets of boundaries. This lower overlap suggests the possibility that music and speech provide complementary boundary evidence to SFX and Mixture.

#### 5.4.2. Audio Separation Limitations

The pretrained MRX model was trained on the Divide and Remaster (DnR) dataset, which was not designed specifically for gameplay audio. Differences between the training data and game audio may affect separation, particularly when music, sound effects, and vocalizations overlap. The estimated stems may share sounds or attenuate event-related cues. When transient sounds are present in multiple stems, their novelty peaks may combine in the weighted audio score and produce spurious boundary candidates. Conversely, attenuated boundary cues may yield weaker novelty peaks that fail to exceed the detection threshold, leading to missed boundaries.

We qualitatively inspected the original mixture and estimated stems in two gameplay excerpts, shown in Figure 6. Each spectrogram displays time horizontally and frequency vertically; brighter colors indicate larger spectral magnitudes on a common decibel scale. In the first excerpt, listening to the separated audio revealed screams and attack sounds in both the estimated SFX and Music stems around 43 s. In the second excerpt, a door sound was audible in both estimated stems around 6 s. These observations illustrate that game sounds may be shared across the separated outputs.

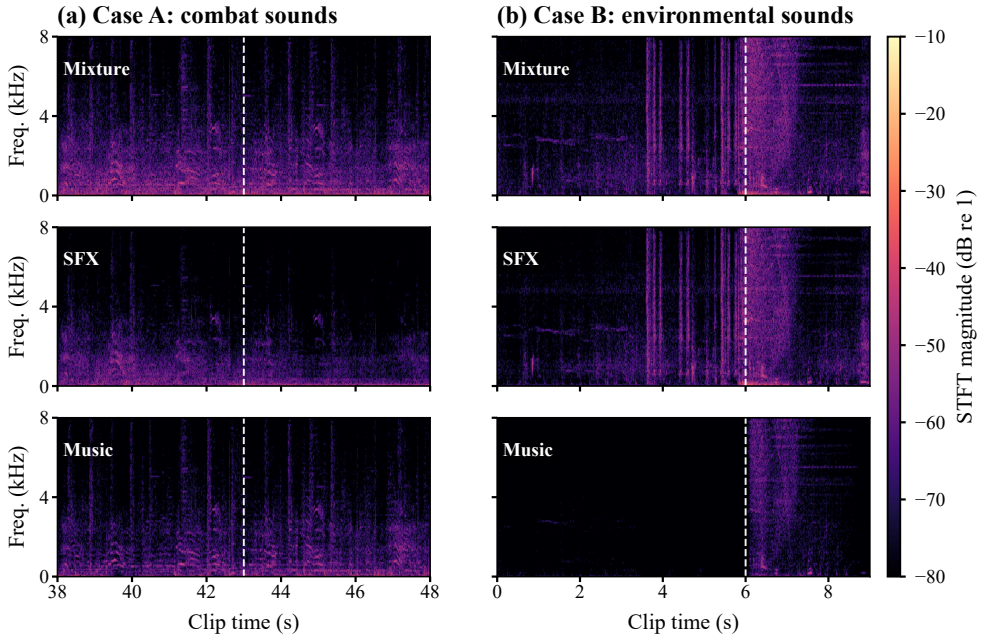



**Figure 6.** Qualitative examples of the mixture and estimated SFX and music stems: (**a**) combat sounds around 43 s; (**b**) a door sound around 6 s. Dashed lines mark these times. Spectrograms use a shared decibel scale; sound descriptions are based on listening.

#### 5.4.3. Overall Method Comparison

The previous single-audio-signal analysis identified SFX as the best-performing single audio signal. We evaluate Audio Signal Fusion to examine whether the potential complementary evidence from music and speech suggested by the overlap analysis provides additional benefit beyond the strongest single audio signal. This section compares SFX, Au-

dio Signal Fusion, the DDM-Net visual baseline, and the proposed Gameplay Audio-Visual Fusion (GAVF) under the same LOGTO protocol.

Table 7 summarizes the F1 results of the main comparison methods across Rel.Dis. thresholds. DDM-Net achieves an Avg. F1 of 0.631, which is lower than the audio-based methods. At the strict Rel.Dis. threshold of 0.05, it achieves 0.304, indicating that accurate boundary localization from visual boundary scores alone remains challenging in gameplay videos. Audio Signal Fusion outperforms SFX only at Rel.Dis. thresholds of 0.05 and 0.10 but achieves lower scores at all remaining thresholds, resulting in an Avg. F1 of 0.693. GAVF achieves an Avg. F1 of 0.720, compared with 0.703 for SFX and 0.693 for Audio Signal Fusion.

**Table 7.** Overall method comparison under LOGTO.

| | Rel.Dis. Threshold | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Method** | **0.05** | **0.10** | **0.15** | **0.20** | **0.25** | **0.30** | **0.35** | **0.40** | **0.45** | **0.50** | **Avg.** |
| SFX | 0.541 | 0.640 | 0.681 | 0.703 | 0.721 | 0.731 | 0.746 | 0.753 | 0.755 | 0.755 | 0.703 |
| Audio Signal Fusion | 0.552 | 0.649 | 0.680 | 0.699 | 0.705 | 0.716 | 0.727 | 0.730 | 0.735 | 0.739 | 0.693 |
| DDM-Net (LOGTO-calibrated) | 0.304 | 0.472 | 0.577 | 0.636 | 0.669 | 0.695 | 0.722 | 0.742 | 0.742 | 0.750 | 0.631 |
| **GAVF** | **0.566** | **0.667** | **0.693** | **0.726** | **0.743** | **0.748** | **0.755** | **0.761** | **0.768** | **0.768** | **0.720** |

Bold scores indicate the highest value in each score column.

Table 8 summarizes the TP and FP counts pooled across the ten held-out folds under LOGTO. Table 7 reports the mean of fold-specific F1 scores, whereas Table 8 reports pooled TP and FP counts across folds. Consequently, F1 scores calculated from these pooled counts and the total number of GT boundaries need not equal the corresponding values in Table 7. Compared with SFX, Audio Signal Fusion slightly increases TP and reduces FP at strict Rel.Dis. thresholds but reduces TP and increases FP at Rel.Dis. 0.25 and 0.50. This suggests that adding music and speech cues does not consistently improve the TP–FP balance over SFX alone. Relative to Audio Signal Fusion, GAVF improves the TP–FP balance by increasing TP across all reported thresholds while keeping FP changes small. Compared with DDM-Net, GAVF increases TP and substantially reduces FP at strict Rel.Dis. thresholds. At looser Rel.Dis. thresholds, DDM-Net matches more GT boundaries but also produces many more FPs, whereas GAVF keeps the FP count lower.

**Table 8.** Diagnostic TP and FP counts under LOGTO. Counts are pooled over the ten held-out folds, where the pooled number of GT boundaries is 316.

| **Method** | **Rel.Dis.** | **TP** | **FP** |
|---|---|---|---|
| SFX | 0.05 | 159 | 106 |
| | 0.10 | 187 | 78 |
| | 0.25 | 210 | 55 |
| | 0.50 | 219 | 46 |
| Audio Signal Fusion | 0.05 | 162 | 104 |
| | 0.10 | 189 | 77 |
| | 0.25 | 206 | 60 |
| | 0.50 | 217 | 49 |
| DDM-Net | 0.05 | 107 | 278 |
| | 0.10 | 165 | 220 |
| | 0.25 | 235 | 150 |
| | 0.50 | 264 | 121 |
| GAVF | 0.05 | 173 | 108 |
| | 0.10 | 200 | 81 |
| | 0.25 | 223 | 58 |
| | 0.50 | 231 | 50 |

To analyze these TP–FP differences, we examine TP-set overlap at the strict Rel.Dis. threshold of 0.05. Figure 7 shows the Jaccard similarity between TP sets of the main

methods. Figure 8 reports the corresponding shared TP counts. GAVF and Audio Signal Fusion share 148 TPs, with a Jaccard similarity of 0.79. This high overlap shows that most GAVF correct detections are also detected by Audio Signal Fusion. Of the 173 GAVF TPs, 25 are unique to GAVF, indicating that GAVF adds correct detections beyond the audio-fusion baseline. GAVF shares 62 TPs with DDM-Net and has a lower Jaccard similarity of 0.28. This shows that GAVF and the visual-only baseline correctly detect largely different boundary sets. These results show that GAVF preserves most of the TP set of Audio Signal Fusion, adds GAVF-only correct detections, and keeps FP changes limited.

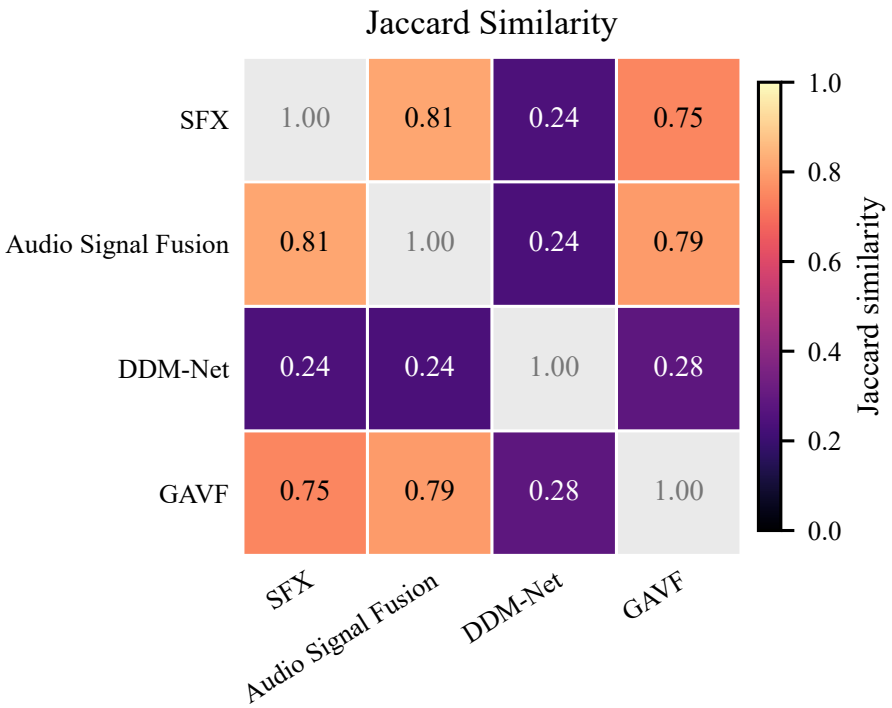



**Figure 7.** Jaccard similarity between TP sets of the main methods under LOGTO at Rel.Dis. 0.05.

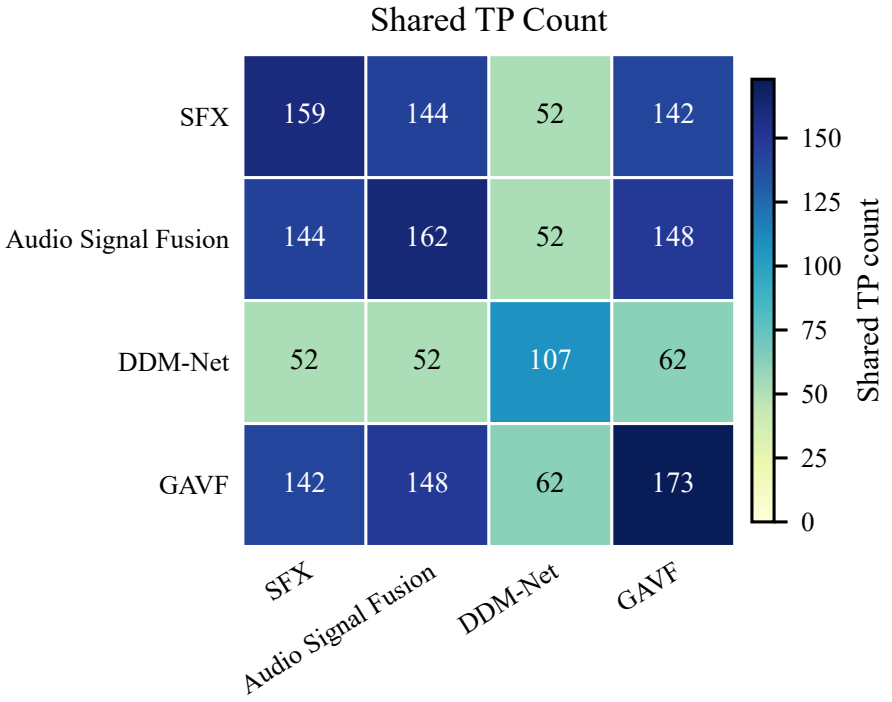



**Figure 8.** Shared TP count between the main methods under LOGTO at Rel.Dis. 0.05.

#### 5.4.4. Fusion Strategy Sensitivity

Weighted summation is used at both fusion stages so that boundary evidence from one signal can contribute even when another signal is weak. In contrast, product fusion suppresses the combined score when an active input approaches zero, which may suppress useful evidence from another signal.

We compared weighted sum with weighted maximum and weighted product at the audio-signal and audio-visual fusion stages. At the audio stage, the maximum takes the largest weighted signal score, and the product multiplies the signal scores raised to their respective weights; signals with zero weight are excluded from the product. At the audio-visual stage, the maximum compares the audio score with the weighted visual score, while the product multiplies the audio score by the visual score raised to its weight. A zero visual weight leaves the audio score unchanged. We also examined the effect of normalizing the aggregated audio curve by its clipwise maximum before audio-visual fusion. The original scale leaves this intermediate curve unchanged; the proposed method uses the original scale and weighted sums at both stages. Each configuration uses the same candidate grids, and its fusion and peak-detection parameters are selected on the other nine sources in each LOGTO fold. We vary one fusion stage at a time relative to the sum–sum baseline.

Table 9 shows that sum–product achieves a higher Avg. F1 than sum–sum on the original score scale (0.749 vs. 0.720). However, a per-game comparison at Rel.Dis. 0.05 shows that sum–product performs better on four games, worse on four, and equally on two. Thus, although product fusion is a promising alternative, its benefit is not consistent across the evaluated gameplay sources.

**Table 9.** Fusion strategy sensitivity under LOGTO. The first operator combines audio signals and the second combines audio and visual scores. Stage normalization divides the aggregated audio curve by its clipwise maximum before audio-visual fusion. F1 values are equally averaged over the ten held-out sources, and Avg. F1 is also averaged over the ten Rel.Dis. thresholds.

| Audio Scale | Audio Operator | AV Operator | F1@0.05 | Avg. F1 |
|---|---|---|---|---|
| Original | Sum | Sum | 0.566 | 0.720 |
| Original | Max | Sum | 0.572 | 0.734 |
| Original | Product | Sum | 0.558 | 0.728 |
| Original | Sum | Max | 0.541 | 0.724 |
| Original | Sum | Product | 0.577 | 0.749 |
| Normalized | Sum | Sum | 0.575 | 0.754 |
| Normalized | Max | Sum | 0.562 | 0.714 |
| Normalized | Product | Sum | 0.554 | 0.734 |
| Normalized | Sum | Max | 0.539 | 0.697 |
| Normalized | Sum | Product | 0.577 | 0.749 |

#### 5.4.5. Effect of Visual Model Selection

To examine visual model selection, we replaced DDM-Net with the Pairwise Boundary Classifier (PC) [8], EfficientGEBD-R50 [10], and DiffGEBD [17], using publicly available GEBD-pretrained checkpoints. PC provides a further local-context comparison, whereas EfficientGEBD-R50 and DiffGEBD provide alternative GEBD approaches with longer input sequences. For each model, the fusion and peak-detection parameters were selected on the other nine recordings in each LOGTO fold; the held-out recording was used only for evaluation. The audio features, ground truth, and evaluation durations were shared across models.

As shown in Table 10, EfficientGEBD-R50 achieves the highest visual-only Avg. F1 of 0.655, whereas DDM-Net yields the highest GAVF Avg. F1 of 0.720. Thus, higher visual-only performance does not necessarily translate to better audio-visual fusion performance. Fusion improves Avg. F1 over the corresponding visual-only result for all four models. However, performance relative to Audio Signal Fusion varies across models: DDM-Net

achieves a higher Avg. F1, DiffGEBD is slightly higher, and PC and EfficientGEBD-R50 are slightly lower. These results show that GAVF can effectively incorporate boundary scores from different GEBD models and improve the corresponding visual-only performance across all four models, demonstrating that the framework is not tied to a specific visual extractor.

**Table 10.** Visual model comparison under LOGTO. Avg. F1 is averaged over the ten Rel.Dis. thresholds.

| Visual Model | Visual Only | | GAVF | |
|---|---|---|---|---|
| | F1@0.05 | Avg. | F1@0.05 | Avg. |
| DDM-Net | 0.304 | 0.631 | 0.566 | 0.720 |
| PC | 0.306 | 0.647 | 0.546 | 0.692 |
| EfficientGEBD-R50 | 0.367 | 0.655 | 0.521 | 0.690 |
| DiffGEBD | 0.249 | 0.621 | 0.532 | 0.695 |
| Audio Signal Fusion | F1@0.05: 0.552; Avg.: 0.693 | | | |

#### 5.4.6. Cross-Dataset Transfer

To examine how parameters tuned on the full gameplay dataset transfer to other video domains, we conduct direct cross-dataset transfer experiments. Unlike the LOGTO evaluation, this transfer experiment applies previously selected gameplay parameter settings without target-domain calibration to the Kinetics-GEBD and TAPOS settings from the GEBD benchmark [8]. The settings differ by method. Annotations in the target datasets are used only for evaluation, and no threshold calibration or model selection is performed using the target datasets. Kinetics-GEBD targets generic event boundaries perceived by humans in general videos, and TAPOS targets sub-action boundaries in sports and action videos. These datasets therefore differ from gameplay videos in annotation criteria and temporal structure. Table 11 summarizes the direct-transfer results for the two datasets.

**Table 11.** Cross-dataset transfer results at Rel.Dis. 0.05 and averaged across the ten Rel.Dis. thresholds. The availability-filtered Kinetics-GEBD subset contains 6840 clips and 31,426 GT boundaries, and the TAPOS subset contains 249 clips and 527 GT boundaries. Scores use representative ground truth and pooled F1; the visual baseline is evaluated on the audio grid.

| Method | Kinetics-GEBD | | TAPOS | |
|---|---|---|---|---|
| | F1@0.05 | Avg. F1 | F1@0.05 | Avg. F1 |
| SFX | 0.136 | 0.248 | 0.172 | 0.387 |
| Audio Signal Fusion | 0.101 | 0.186 | 0.161 | 0.370 |
| DDM-Net | **0.198** | **0.271** | **0.205** | 0.379 |
| GAVF | 0.168 | 0.262 | 0.188 | **0.404** |

Bold scores indicate the highest value in each score column.

With the fixed gameplay settings, GAVF obtains an Avg. F1 of 0.262 on the Kinetics-GEBD subset, below the visual-only score of 0.271. On the TAPOS subset, GAVF obtains 0.404 versus 0.379 for the visual-only score. The corresponding F1 at Rel.Dis. 0.05 is 0.168 versus 0.198 on Kinetics-GEBD and 0.188 versus 0.205 on TAPOS. Thus, the relative benefit of fusion depends on the target and tolerance. The gameplay and transfer evaluations use different boundary definitions and aggregation procedures: the former averages held-out source scores, while the latter pools counts across the available target clips. The target subsets and evaluation protocol also differ from the native benchmark settings. Two Kinetics-GEBD clips have empty audio grids and yield no predictions for any method. The numerical gap alone therefore cannot isolate the causes of transfer errors.

To examine post-processing sensitivity, we additionally evaluated a prespecified diagnostic that jointly changes the threshold coefficient from 1.4 to 0 and the minimum peak distance from 14 to 1 score sample, without selecting these values using target annotations. At Rel.Dis. 0.05, the Kinetics-GEBD true-positive count increases from 3083 to 10,647, while

false positives increase from 2243 to 9046. On TAPOS, true positives increase from 90 to 288, but false positives increase from 340 to 2785 and Avg. F1 decreases from 0.404 to 0.239. Thus, the same relaxation has different effects across target datasets.

Three examples illustrate these effects. Figure 9a shows a Kinetics-GEBD clip in which a person repeatedly raises and lowers their arms. The 13 annotated boundaries occur throughout this repeated motion; the original GAVF setting produces no predictions, while the relaxed setting matches six boundaries at Rel.Dis. 0.05. Figure 9b shows a TAPOS athletics clip in which the camera follows the field after an athlete performs a throwing motion. The two annotated boundaries occur at 13.47 and 13.73 s; the original setting produces no predictions, whereas the relaxed setting produces seven false positives and no matches. In another Kinetics-GEBD clip showing an aerial view during paragliding, the visual-only predictions at 0.5 and 8.5 s match two of three boundaries, while the original GAVF prediction at 4.5 s matches none, as shown in Figure 10. Times are relative to each clip. These examples were selected after evaluation to illustrate error types, not to estimate performance or establish their cause.

These examples show that transfer results are sensitive to the post-processing settings selected on gameplay data. Differences in boundary definitions and acoustic cues may also affect transfer performance, but their individual contributions cannot be determined from the present experiments.

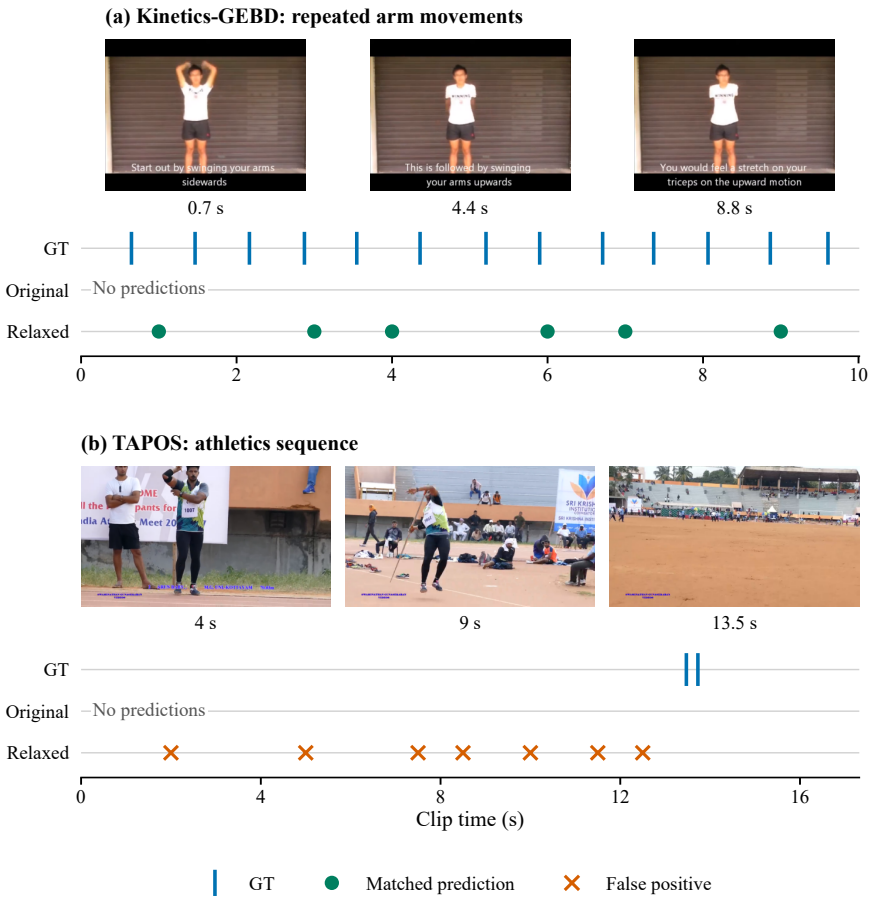



**Figure 9.** Cross-dataset examples under the original and relaxed post-processing settings: (**a**) repeated arm movements in a Kinetics-GEBD clip; (**b**) an athletics sequence in a TAPOS clip. Markers indicate ground-truth boundaries, matched predictions, and false positives at Rel.Dis. 0.05.

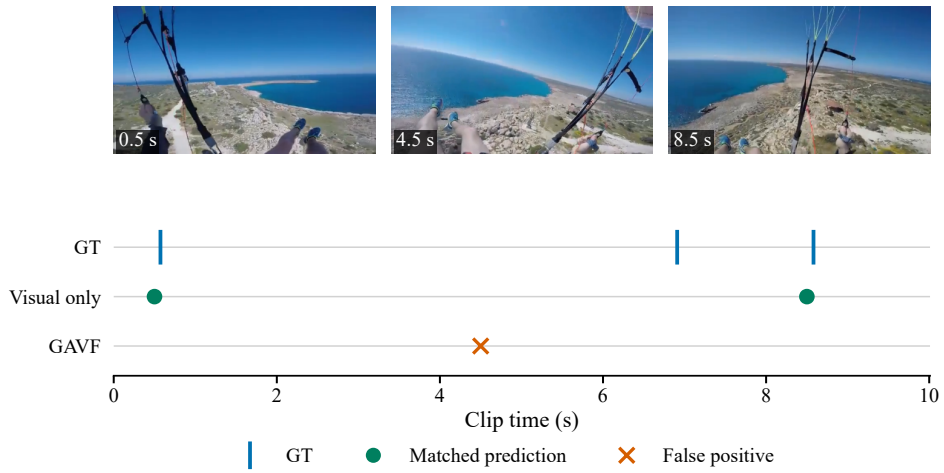



**Figure 10.** A selected Kinetics-GEBD clip showing representative frames, ground-truth (GT) boundaries, and visual-only and GAVF prediction timelines. Markers indicate matched predictions and false positives at Rel.Dis. 0.05. Each method uses its respective fixed gameplay settings.

## 6. Related Work

Gameplay video analysis has used both internal game data and observable public-video signals to infer play context. Telemetry-based studies can analyze player behavior from structured in-game data [4,5], but such information is not available for arbitrary public gameplay videos. Public-video-based approaches have used heuristic HUD analysis, game-specific sound cues, or external signals such as chat and viewer reactions to infer play situations or detect highlights [2,3,18]. These approaches demonstrate the value of gameplay-video analysis, but their reliance on telemetry, game-specific templates, or external streaming metadata limits their applicability to unseen game titles and arbitrary public gameplay videos. For semantic boundary detection in public gameplay videos, the relevant comparison is therefore temporal video structuring from observable audio and visual signals.

Temporal video structuring research has mainly organized videos around edit points, action labels, chapters, or multimodal event segments. These targets are useful for general video organization, but they differ from gameplay-state transitions in public gameplay videos. Shot boundary detection identifies edit-level visual discontinuities such as cuts and fades [19,20], whereas gameplay-state transitions can occur without visible edit points. Temporal action segmentation partitions videos into intervals with predefined action labels [21], while gameplay-state transitions are boundary points rather than labeled action intervals. Video chaptering organizes long videos into high-level semantic units using textual or multimodal cues [22], but chapter-level units are typically coarser than gameplay-state boundaries. Multimodal event localization and long-form video benchmarks use audio, visual, and language cues to identify temporal events [23,24], but they typically focus on event segments or labeled semantic units rather than boundary times for gameplay-state transitions.

Among these directions, Generic Event Boundary Detection (GEBD) provides the closest point of comparison for this study because it focuses on boundary times rather than labeled action intervals or coarse chapters. GEBD predicts human-perceived event boundaries without relying on a fixed action taxonomy. Recent GEBD methods improve boundary detection in general videos through stronger visual modeling [9–11]. However, existing GEBD datasets and models mainly target general videos rather than gameplay-specific temporal structure. They therefore do not directly cover boundary conditions caused by UI-heavy motion, repetitive combat patterns, and weak alignment between

visual changes and gameplay-state transitions. These differences make gameplay-state transitions difficult to capture with general-purpose temporal video structuring or GEBD models alone.

Audio analysis provides another line of related work for detecting temporal changes. Classical audio segmentation methods represent signals using spectral features such as MFCCs [12] and identify novelty from changes in local self-similarity [13]. Deep learning-based Sound Event Detection (SED) estimates the presence and onset/offset times of sound events [25,26]. However, SED typically assumes predefined event categories, whereas gameplay videos vary widely in sound design across titles. This category dependence limits the direct applicability of SED to semantic boundary detection in public gameplay videos.

## 7. Conclusions

We proposed and evaluated Gameplay Audio-Visual Fusion (GAVF), a score-level fusion framework for semantic boundary detection in public gameplay videos. The framework is designed to detect gameplay-state transitions that are not reliably captured by a single cue, because visual changes may be weakly aligned with state transitions and audio cues may be noisy or incomplete. It estimates source-separated audio novelty scores, aligns and fuses them with a visual boundary score, and generates boundary candidates from the fused audio-visual score. To evaluate this framework, we constructed a dataset of 123 clips from 10 game titles with 316 majority-agreed GT boundaries.

The results indicate that audio cues provide boundary evidence in this domain. At a Rel.Dis. threshold of 0.05, GAVF achieves an F1 score of 0.566, compared with 0.552 for Audio Signal Fusion and 0.304 for the calibrated DDM-Net visual baseline. These results suggest that incorporating visual cues alongside audio novelty scores can provide additional evidence for gameplay-state boundary detection. Cross-dataset evaluation showed that audio-visual fusion did not consistently outperform the visual-only baseline across the Kinetics-GEBD and TAPOS subsets. These results suggest that applying GAVF to boundary types beyond gameplay-state transitions requires a separate validation of the fusion and post-processing settings.

Future work will extend the dataset to cover a wider variety of game genres and channels, explore pretrained audio embeddings in addition to MFCC-based novelty, and compare the framework with visual models adapted to gameplay videos.

**Author Contributions:** Conceptualization, S.L. and H.S.; methodology, S.L.; software, S.L.; validation, S.L.; formal analysis, S.L.; investigation, S.L.; resources, H.S.; data curation, S.L.; writing—original draft preparation, S.L.; writing—review and editing, H.S.; visualization, S.L.; supervision, H.S.; project administration, H.S.; funding acquisition, H.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by a 2025 Research Grant from Sangmyung University (2025-A000-0326).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The original gameplay videos are not redistributed due to copyright restrictions. The dataset metadata, annotations, evaluation splits, and evaluation scripts are available on request from the corresponding author.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Appendix A. Selected Parameters in Each LOGTO Fold

Fold identifiers are consistent across Tables A1 and A2. Each fold holds out all clips from the corresponding game title for testing and uses the remaining nine game titles for parameter selection. The fold-to-title mapping is listed below.

| Fold | Held-Out Game Title |
|---|---|
| 1 | Diablo IV |
| 2 | Genshin Impact |
| 3 | Assassin's Creed Odyssey |
| 4 | NieR: Automata |
| 5 | Horizon Forbidden West |
| 6 | Lies of P |
| 7 | Black Myth: Wukong |
| 8 | Elden Ring |
| 9 | Monster Hunter Rise |
| 10 | Crimson Desert |

Table A1 reports the selected post-processing parameters. Table A2 reports the corresponding fusion weights.

**Table A1.** Post-processing parameters selected in each LOGTO fold. Each entry denotes $(k, d_{\text{sec}})$, where $d_{\text{sec}}$ is measured in seconds.

| Fold | Mixture | Music | SFX | Speech | Audio Signal Fusion | DDM-Net | GAVF |
|---|---|---|---|---|---|---|---|
| 1 | (0.7, 7) | (0.2, 10) | (0.4, 10) | (0.0, 7) | (0.8, 7) | (2.1, 15) | (1.3, 7) |
| 2 | (0.9, 7) | (0.2, 10) | (0.4, 10) | (0.0, 7) | (0.7, 7) | (2.2, 15) | (1.4, 7) |
| 3 | (0.5, 7) | (0.2, 7) | (0.4, 10) | (0.0, 7) | (0.6, 7) | (1.8, 15) | (1.1, 7) |
| 4 | (0.5, 7) | (0.4, 7) | (0.4, 10) | (0.0, 7) | (0.8, 7) | (1.8, 15) | (1.1, 7) |
| 5 | (0.7, 7) | (0.2, 10) | (0.4, 10) | (0.0, 7) | (0.7, 7) | (2.1, 15) | (1.3, 7) |
| 6 | (0.5, 7) | (0.1, 10) | (0.4, 10) | (0.0, 7) | (0.7, 7) | (2.1, 15) | (1.3, 7) |
| 7 | (0.9, 1.5) | (0.2, 10) | (0.4, 10) | (0.0, 7) | (0.7, 7) | (1.8, 15) | (1.1, 7) |
| 8 | (0.8, 7) | (0.3, 7) | (0.4, 10) | (0.0, 7) | (0.7, 7) | (2.4, 15) | (1.3, 7) |
| 9 | (0.5, 7) | (0.1, 10) | (0.1, 10) | (0.0, 7) | (0.7, 7) | (2.2, 15) | (1.1, 7) |
| 10 | (0.5, 7) | (0.1, 10) | (0.4, 10) | (0.0, 7) | (0.7, 7) | (2.1, 15) | (1.3, 7) |

$k$ is the dimensionless coefficient of the adaptive threshold $\mu + k\sigma$. The minimum-distance parameter $d$ is measured in score samples and reported here in seconds as $d_{\text{sec}} = d\Delta t$, where $\Delta t = 0.5$ s for audio-based methods and GAVF and 0.05 s for DDM-Net.

**Table A2.** Fusion weights selected in each LOGTO fold.

| Fold | Audio Signal Fusion | GAVF Audio Fusion Weights | GAVF $w_{\text{DDM}}$ |
|---|---|---|---|
| 1 | (0.75, 0.25, 0.75, 0.00) | (0.75, 0.50, 0.75, 0.25) | 0.75 |
| 2 | (0.50, 0.25, 0.75, 0.25) | (0.75, 0.50, 1.00, 0.25) | 1.25 |
| 3 | (0.75, 0.25, 1.00, 0.25) | (0.50, 0.50, 1.25, 0.00) | 1.00 |
| 4 | (0.75, 0.50, 1.25, 0.25) | (0.50, 0.50, 1.25, 0.00) | 1.00 |
| 5 | (0.50, 0.25, 0.75, 0.25) | (0.75, 0.50, 0.75, 0.25) | 0.75 |
| 6 | (0.75, 0.25, 1.25, 0.50) | (0.75, 0.25, 0.75, 0.00) | 1.00 |
| 7 | (0.50, 0.25, 0.75, 0.25) | (0.50, 0.50, 1.25, 0.00) | 1.00 |
| 8 | (0.50, 0.50, 0.75, 0.00) | (0.25, 0.25, 0.75, 0.00) | 0.75 |
| 9 | (0.50, 0.25, 0.75, 0.25) | (0.50, 0.50, 1.25, 0.00) | 1.00 |
| 10 | (0.50, 0.25, 0.75, 0.25) | (0.75, 0.50, 0.75, 0.25) | 0.75 |

Audio fusion weight tuples are ordered as $(w_{\text{Mix}}, w_{\text{Music}}, w_{\text{SFX}}, w_{\text{Speech}})$. Each method's fusion weights are selected jointly with its fold-specific $k$ and $d$ parameters. The weights are not constrained to sum to one.

Table A3 lists the candidate parameter values used for the validation-based grid search described in Section 5.2.

**Table A3.** Candidate parameter values used for validation-based selection in each LOGTO fold.

| Parameter | Method | Candidate Values |
|---|---|---|
| $k$ | Single audio signal methods | $\{0, 0.1, \ldots, 2.0\}$ |
| $k$ | Audio Signal Fusion, GAVF | $\{0.6, 0.7, \ldots, 2.2\}$ |
| $k$ | DDM-Net | $\{0, 0.1, \ldots, 2.5\}$ |
| $d_{\text{sec}}$ | Single audio signal methods | $\{1.5, 2.5, 4, 5, 7, 10\}$ |
| $d_{\text{sec}}$ | Audio Signal Fusion, GAVF | $\{2.5, 4, 5, 7, 10\}$ |
| $d_{\text{sec}}$ | DDM-Net | $\{3, 4.5, 6, 7.5, 9, 12, 15\}$ |
| $w_{\text{Mix}}$ | Audio Signal Fusion, GAVF | $\{0.25, 0.50, 0.75\}$ |
| $w_{\text{Music}}$ | Audio Signal Fusion, GAVF | $\{0.25, 0.50, 0.75\}$ |
| $w_{\text{SFX}}$ | Audio Signal Fusion, GAVF | $\{0.75, 1.00, 1.25\}$ |
| $w_{\text{Speech}}$ | Audio Signal Fusion, GAVF | $\{0, 0.25, 0.50\}$ |
| $w_{\text{DDM}}$ | GAVF | $\{0, 0.025, 0.05, 0.1, 0.2, 0.25, 0.3, 0.5, 0.75, 1.0, 1.25, 1.5\}$ |

$d_{\text{sec}}$ is the minimum distance between selected peaks expressed in seconds, converted from score samples using the nominal 0.5 s interval of the audio novelty scores for audio-based methods and GAVF, and the native 0.05 s interval of the visual boundary scores for DDM-Net.

## References


1. Zhang, X.; Zhan, Z.; Holtz, M.; Smith, A.M. Crawling, indexing, and retrieving moments in videogames. In Proceedings of the 13th International Conference on the Foundations of Digital Games, Malmö, Sweden, 7–10 August 2018; pp. 1–10.
2. Ringer, C.; Nicolaou, M.A. Deep unsupervised multi-view detection of video game stream highlights. In Proceedings of the 13th International Conference on the Foundations of Digital Games, Malmö, Sweden, 7–10 August 2018; pp. 1–6.
3. Park, G.M.; Hyun, H.I.; Kwon, H.Y. Multimodal learning model based on video–audio–chat feature fusion for detecting e-sports highlights. *Appl. Soft Comput.* **2022**, *126*, 109285.
4. Su, Y.; Backlund, P.; Engström, H. Comprehensive review and classification of game analytics. *Serv. Oriented Comput. Appl.* **2021**, *15*, 141–156.
5. Gagné, A.R.; El-Nasr, M.S.; Shaw, C.D. A Deeper Look at the use of Telemetry for Analysis of Player Behavior in RTS Games. In *Proceedings of the International Conference on Entertainment Computing*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 247–257.
6. Raad, M.A.; Ahuja, A.; Barros, C.; Besse, F.; Bolt, A.; Bolton, A.; Brownfield, B.; Buttimore, G.; Cant, M.; Chakera, S.; et al. Scaling instructable agents across many simulated worlds. *arXiv* **2024**, arXiv:2404.10179.
7. Choi, S.; Jung, J.; Seong, H.; Kim, M.; Kim, M.; Cho, Y.; Kim, Y.; Park, Y.; Yu, Y.; Lee, Y. D2E: Scaling Vision-Action Pretraining on Desktop Data for Transfer to Embodied AI. *arXiv* **2025**, arXiv:2510.05684.
8. Shou, M.Z.; Lei, S.W.; Wang, W.; Ghadiyaram, D.; Feiszli, M. Generic event boundary detection: A benchmark for event segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 8075–8084.
9. Tang, J.; Liu, Z.; Qian, C.; Wu, W.; Wang, L. Progressive attention on multi-level dense difference maps for generic event boundary detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 3355–3364.
10. Zheng, Z.; Zhang, Z.; Wang, Y.; Song, S.; Huang, G.; Yang, L. Rethinking the architecture design for efficient generic event boundary detection. In Proceedings of the 32nd ACM International Conference on Multimedia, Melbourne, VIC, Australia 28 October–1 November 2024; pp. 1215–1224.
11. Gothe, S.V.; Agarwal, V.; Ghosh, S.; Vachhani, J.R.; Kashyap, P.; Raja, B.R.K. What's in the Flow? Exploiting Temporal Motion Cues for Unsupervised Generic Event Boundary Detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2024; pp. 6941–6950.
12. Davis, S.; Mermelstein, P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoust. Speech, Signal Process.* **1980**, *28*, 357–366. https://doi.org/10.1109/TASSP.1980.1163420.
13. Foote, J. Automatic audio segmentation using a measure of audio novelty. In Proceedings of the 2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532), New York, NY, USA, 30 July–2 August 2000; Volume 1, pp. 452–455. https://doi.org/10.1109/ICME.2000.869637.
14. Kay, W.; Carreira, J.; Simonyan, K.; Zhang, B.; Hillier, C.; Vijayanarasimhan, S.; Viola, F.; Green, T.; Back, T.; Natsev, P.; et al. The kinetics human action video dataset. *arXiv* **2017**, arXiv:1705.06950.
15. Shao, D.; Zhao, Y.; Dai, B.; Lin, D. Intra- and Inter-Action Understanding via Temporal Action Parsing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 730–739.
16. Petermann, D.; Wichern, G.; Wang, Z.Q.; Le Roux, J. The cocktail fork problem: Three-stem audio separation for real-world soundtracks. In *Proceedings of the ICASSP 2022—IEEE International Conference on Acoustics, Speech and Signal Processing*; IEEE: New York, NY, USA, 2022; pp. 526–530.

17. Hwang, J.; Gong, D.; Kim, M.; Cho, M. Generic Event Boundary Detection via Denoising Diffusion. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Honolulu, HI, USA, 19–23 October 2025; pp. 14084–14094.
18. Marczak, R.; Schott, G.; Hanna, P. Postprocessing gameplay metrics for gameplay performance segmentation based on audiovisual analysis. *IEEE Trans. Comput. Intell. AI Games* **2015**, *7*, 279–291.
19. Koprinska, I.; Carrato, S. Temporal video segmentation: A survey. *Signal Process. Image Commun.* **2001**, *16*, 477–500.
20. Kar, T.; Kanungo, P.; Mohanty, S.N.; Groppe, S.; Groppe, J. Video shot-boundary detection: Issues, challenges and solutions. *Artif. Intell. Rev.* **2024**, *57*, 104.
21. Ding, G.; Sener, F.; Yao, A. Temporal action segmentation: An analysis of modern techniques. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 1011–1030.
22. Yang, A.; Nagrani, A.; Laptev, I.; Sivic, J.; Schmid, C. Vidchapters-7m: Video chapters at scale. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 49428–49444.
23. Tian, Y.; Shi, J.; Li, B.; Duan, Z.; Xu, C. Audio-visual event localization in unconstrained videos. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 252–268.
24. Geng, T.; Zhang, J.; Wang, Q.; Wang, T.; Duan, J.; Zheng, F. Longvale: Vision-audio-language-event benchmark towards time-aware omni-modal perception of long videos. In Proceedings of the Computer Vision and Pattern Recognition Conference, Nashville, TN, USA, 11-15 June 2025; pp. 18959–18969.
25. Phan, H.; Chen, O.Y.; Koch, P.; Pham, L.; McLoughlin, I.; Mertins, A.; De Vos, M. Unifying isolated and overlapping audio event detection with multi-label multi-task convolutional recurrent neural networks. In *Proceedings of the ICASSP 2019—IEEE International Conference on Acoustics, Speech and Signal Processing*; IEEE: New York, NY, USA, 2019; pp. 51–55.
26. Yoshinaga, T.; Tanaka, K.; Bando, Y.; Imoto, K.; Morishima, S. Onset-and-offset-aware sound event detection via differentiable frame-to-event mapping. *IEEE Signal Process. Lett.* **2025**, *32*, 186–190.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.