RESEARCH PAPER

Logical Operators for Multimodal Fusion in Temporal Video Scene Segmentation

Letícia B. Barbosa [Universidade de São Paulo - Escola de Engenharia de São Carlos | *leticiabarbosa@usp.br*]

Rudinei Goularte [Universidade de São Paulo - Instituto de Ciências Matemáticas e de Computação | *rudinei@icmc.usp.br*]

Abstract. Early fusion techniques in content analysis aim to enhance efficacy by generating compact data models that retain semantic clues from multimodal data. Initial attempts used fusion operators at low-level feature space, which compromised data representativeness. This led to the development of complex operations inseparable from multimodal semantic clues processing. Previous studies showed that simple arithmetic-based operators could be as effective as complex operations when applied at the mid-level feature space, highlighting an unexplored opportunity to assess the efficacy of logical operators. This paper investigates the application of logical fusion operators (And, Or, Xor) at the mid-level feature space for Temporal Video Scene Segmentation. Comparative analysis demonstrates that Or and Xor logical operators are viable alternatives in the specific Temporal Video Scene Segmentation content analysis tasks.

Keywords: Multimodal Fusion, Fusion Operators, Video Scene Segmentation, Video Analysis

1 Introduction

Videos have become ubiquitous in modern life, appearing in news, streaming platforms, and social media. While this proliferation expands access to knowledge, it also worsens the information overload—the difficulty of identifying relevant content amid vast amounts of data (Gross [1965]). Addressing this issue has driven research in Multimedia Content Analysis, where multimodal approaches integrate data from different sources (modalities) to enhance tasks such as search (Wei *et al.* [2024]), navigation (Ashutosh *et al.* [2024]), summarization (Jangra *et al.* [2023]), and segmentation (Tan *et al.* [2024]).

A key challenge in these tasks is the semantic gap (Smeulders *et al.* [2000]), where raw data lacks the necessary context for accurate interpretation. To mitigate this, multimodal fusion techniques combine different modalities (e.g., visual and aural (aka audio)) into a unified representation, reducing data volume while preserving meaningful distinctions. Early fusion, which integrates modalities at an initial stage, allows better correlation exploration and reduces computational costs by running tasks only once. However, traditional early fusion methods—such as concatenation and simple arithmetic operators (summation, maximum, and average) directly at feature vectors level—struggle with issues like data heterogeneity, differing dimensionalities, and synchronization misalignment (Snoek *et al.* [2005]).

To overcome these limitations, mid-level feature representations enrich low-level data with semantic context, improving efficacy (Jhuo *et al.* [2014]). Prior studies have demonstrated that simple arithmetic fusion operators can achieve reasonable efficacy compared to more complex, computationally expensive methods (Beserra *et al.* [2020]; Gôlo *et al.* [2024]). However, logical operators (And, Or and Xor) remain unexplored.

This paper investigates the use of logical operators for multimodal fusion at the mid-level feature representation space, specifically within the Temporal Video Scene Segmentation (TVSS) task. The TVSS pipeline was modified only in its fusion module, ensuring that differences in segmentation results were solely due to the chosen fusion operator. Comparisons were made between multimodal representations fused with arithmetic and logical operators, as well as monomodal (visual or aural) baselines. The results indicate that Or and Xor operators are viable alternatives for TVSS and may be applicable to other multimedia content analysis tasks.

The remainder of this paper is structured as follows: Section 2 reviews related work on multimodal fusion; Section 3 defines the arithmetic and logical fusion operators; Section 4 describes the TVSS pipeline, detailing feature extraction, the low-level to mid-level features process, fusion strategies, and the segmentation algorithm; Section 5 presents evaluation methods and results; and Section 6 concludes with final remarks.

2 Related Work

Multimodal early data fusion integrates information from multiple sources by combining features from different modalities into a single feature vector. As Snoek *et al.* [2005] discussed, initial early fusion approaches directly concatenated low-level feature vectors, which, while straightforward, doubled the data volume. Nowadays, the literature reports various approaches utilizing neural network models for data fusion (Liu *et al.* [2022], Xing *et al.* [2024], Pereira *et al.* [2024], Jia and Lao [2022]).

In the specific field of video scene segmentation, state-of-the-art works include Xing *et al.* [2024], which fuses feature embeddings from video transcripts and video frames by means of a cross-modal attention mechanism in order to temporally segment videos into topics/scenes. Also includes VSMBD (Tan *et al.* [2024]), which employs a visual multi-feature self-supervised learning method to model upon large-scale pre-trained visual encoders, extracting foreground and background visual features. This model is then applied

in the temporal video scene segmentation. Another example is TransNet V2 (Soucek and Lokoc [2024]), which segments videos into scenes by concatenating visual inputs from RGB color histograms and proposed convolutional DDCNN-based deep architecture features. Those examples highlight that the fusion process is inherently a model decision, grappling with AI explainability and limiting the number of input modalities.

On the other hand, non-deep learning early fusion methods are more explainable, generally have no limits on input types, and can reduce data volume effectively. These methods fall into two categories: those based on finding correlations among modalities and those applying early fusion operators. The first group faces the drawback of task-specific fusion, making reuse difficult. For example, Yang *et al.* [2022] focused on multimodal fusion using the Laplacian matrix and medium-level semantic features from visual and textual modalities involving hypergraph construction with high computational cost. Jia and Lao [2022] concentrated on medical image fusion using MRI images transformed by regional homogeneity, achieving fusion via Canonical Correlation Analysis (CCA) with a kernel function. However, this technique is not readily extendable to other domains. Samadiani et al. Samadiani *et al.* [2022] investigated emotion recognition using medium and low-level semantic features fusing visual and aural modalities through a sparse representation matrix, but the technique is limited to emotion classification.

The second group applies simple mathematical operators to feature vectors from different modalities, resulting in a fused vector with the same dimensionality as the input ones (Beserra *et al.* [2020]; Samadiani *et al.* [2022]). Previous work (Beserra and Goularte [2023]) proved that mid-level feature space fusion using these operators are effective and simpler in processing than more complex fusion methods, including those based on deep learning. However, this group just explored arithmetic operators, leaving a gap for exploring the efficacy of logical fusion operators (*And*, *Or*, *Xor*). Logical operators share mathematical operators simplicity and explainability, and are faster (Patterson and Hennessy [2021]). Thus, if logical operators can be proven to deliver comparable or superior efficacy to mathematical operators, their utilization becomes advantageous for tasks such as temporal video scene segmentation.

3 Early Fusion Operators in the Mid-Level Feature Space

As defined by Beserra and Goularte [2023], an early fusion operator is a mathematical arrangement of procedures with a set of multiple feature vectors as input that outputs a single feature vector representing this set. This representation should have a lower data volume than the set and keep semantics, as it will be used as input to the following stages of a task's pipeline. In the mid-level feature space, the feature vectors are generally frequency histograms. Each bin results from computations over a set of low-level descriptors, enriching the final representation with semantic information extracted from the original feature vectors (see Section 4.2).

To define fusion space and operators, lowercase letters with subscript numbers represent a histogram, like h_1

and h_2, and uppercase letters represent the resulting vector/histogram from the early fusion operators: $H = h_1 \, op \, h_2$. Each bin is represented by its index, so h_{12}, for example, represents the second pattern counting in the histogram h_1 representation. The fusion space is the matrix M_{nk}, where each line n stands for a k dimensional histogram. In the cases where a histogram has a lower dimension, $Inputation$ techniques should be applied, inserting data (zeros, for instance) to fill the gap until the correct dimension.

In this way, operators can be applied to each column of the feature space M, and the arithmetic ones of interest in this work are given by Equations 1, 2 and 3 as defined by Beserra and Goularte [2023]:

$$H_{Sum} = [\sum_{i=1}^{n} h_{i1}, \sum_{i=1}^{n} h_{i2}, \cdots, \sum_{i=1}^{n} h_{ik}] \quad (1)$$

where the Sum operation consists of a sum of each bin in the histograms. Instead of increasing dimensionality, this operator raises the final signal amplitude representation. High-valued signals existing in one or more operands may increase bias in favor of those signals, even when there are low-valued signals in the other operands. This could be useful to distinguish between relevant signals and noise.

$$H_{\text{Max}} = \left[\max_{i=1}^{n} h_{i1}, \max_{i=1}^{n} h_{i2}, \cdots, \max_{i=1}^{n} h_{ik}\right] \quad (2)$$

where the Max operation consists of keeping the maximum value of each bin in the histograms (each M's column). The rationale for this operator is that high-valued signals determine the resulting representation. This could be useful to represent the dominant values in the operands. However, unlike Sum, if some dominant signals are noise, they will persist only in the fused representation. This operator does not need any imputation process where the histograms' dimensionalities are different. However, higher-dimensional histograms may influence most of the final result since lower-dimensional histograms are not compared in the last columns.

$$H_{\text{Avg}} = [avg_{i=1}^{n} h_{i1}, avg_{i=1}^{n} h_{i2}, \cdots, avg_{i=1}^{n} h_{ik}] \quad (3)$$

where the Avg operation consists of calculating the average value for each M's column. The rationale here is that the average acts like a smoothing filter, which could help when very high or very low signals are noise.

Regarding logical operators, operations are also done on the columns of M. The rationale is not obvious since logical operations are made bit-wise, resulting in zeros or ones. However, generally, we may think the Or operator behaves like an approximation of a Sum, and the And operator behaves like a bit dot product approximation. This may lead us to believe the Or operator is more advantageous for feature fusion than the arithmetic operators since logical operations are faster and Or behaves like Sum [1].

---

[1] Sum achieves better arithmetic operators' results in the experiments (Subsection 5.3).

However, the Xor operator gives positive (ones) when the operands are different. So, there is a chance, in this case, that the operands' information is complementary, and complementarity is a key concept when thinking about multimodality [2]. Of course, if the operands are equal, there is no complementarity, and the final representation will be zeros. We provide some answers in Section 5.3. The formulation for early fusion logical operators is given by Equations 4, 5 and 6:

$$H_{\text{and}} = [\text{and}_{i=1}^n h_{i1}, \text{ and}_{i=1}^n h_{i2}, \cdots, \text{ and}_{i=1}^n h_{ik}] \quad (4)$$

$$H_{\text{or}} = [\text{or}_{i=1}^n h_{i1}, \text{ or}_{i=1}^n h_{i2}, \cdots, \text{ or}_{i=1}^n h_{ik}] \quad (5)$$

$$H_{\text{xor}} = [\text{xor}_{i=1}^n h_{i1}, \text{ xor}_{i=1}^n h_{i2}, \cdots, \text{ xor}_{i=1}^n h_{ik}] \quad (6)$$

4 TVSS Pipeline

In this Section, we present the TVSS pipeline and its modules, how the mid-level feature representations are generated, how they are fused, and the TVSS algorithm used to predict the scene boundaries. This pipeline is used in the Section 5 comparative analysis between logical and arithmetic simple operators aiming to verify the efficacy of fusion logical operators at mid-level feature space. The TVSS task aims to predict all scene boundaries of a given input video automatically. A scene is defined as a sequence of semantically correlated adjacent video shots, which, in turn, are contiguous sequences of frames captured by a single camera (Koprinska and Carrato [2001]). According to Beserra and Goulart [2023], TVSS is suitable for the proposed analysis because: it is a common preprocessing step for most of the multimedia content analysis tasks; it is a typical example of a video analysis task needing to reduce the data volume of frames and shots by alternative representations (features vectors and features histograms); those representations typically benefit from multimodal fusion.
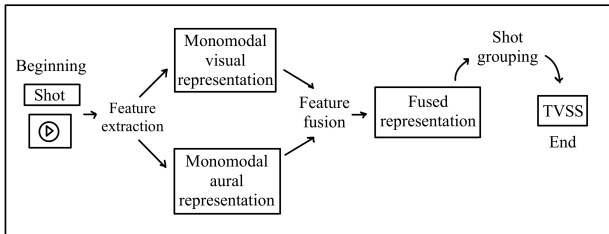

Figure 1. TVSS Pipeline.

Figure 1 illustrates the TVSS pipeline. From the left to the right, it begins with the set of shots of the input video ($Shot$ labeled box). From each shot, the $'Feature\ Extraction'$ module proceeds low-level feature

---

[2]Important data helping to represent information that may be present on a modality but not on others, making the joint use of modalities better Snoek *et al.* [2005]

vectors extraction (Subsection 4.1) and enrichment, turning those vectors into mid-level feature histograms (Subsection 4.2) representing the shots. This process is made for each modality, resulting in monomodal representations. This work used two different modalities (visual and aural), as depicted in the two $Monomodal$ Figure 1 labeled boxes. Furthermore, the pipeline has no restrictions on the number of modalities or the number of different features.

Those mid-level monomodal representations are then fused by the $'Feature\ fusion'$ module, using one of the early fusion operators described in Section 3, outputting fused multimodal shot representations ($Fused\ representation$ labeled box). Next, the fused shot representations are grouped into scenes by the $'Shot\ grouping'$ module, which uses a video scene segmentation algorithm (Subsection 4.4). The result is a list of shots classified as scene boundaries ($TVSS$ labeled box).

4.1 Feature Extraction

Taking the visual case as an example (the same applies to the aural case), the Feature Extraction process (Figure 1) extracts visual low-level feature vectors from the video dataset. A keyframe (the middle frame) was chosen for the visual modality to represent each shot. This is commonly practiced to extract visual information from videos (Jhuo *et al.* [2014]; Beserra *et al.* [2020]; Baraldi *et al.* [2015]). The pipeline does not impose restrictions on the kind of features to be extracted, which can be either hand-crafted or deep features. In this work, we extracted well-known SIFT (Scale Invariant Feature Transform, proposed by Lowe [2004]) visual features from each keyframe.

Aural features, in turn, were extracted using MFCC (Mel-Frequency Cepstrum Coefficients), a widely recognized handcrafted method for representing speech patterns as feature vectors. This approach is well-suited to the BBC dataset and multimodality, as the BBC content often includes a narrator discussing concepts that may have corresponding visual elements. For the extraction process we applied the approach proposed by Beserra *et al.* [2020], where the whole aural content of a shot is used since the aural data volume is considerably smaller than the visual one. The MFCC descriptors were obtained from the aural stream divided into 30 ms frames with a 10 ms overlapped window. This approach has proven satisfactory in speech recognition applications (Sen *et al.* [2019]). SIFT and MFCC features were extracted using the implementations provided by the OpenCV Python API.

4.2 Mid-Level Features Representation

Next in the pipeline, this section describes the process of turning the extracted visual and aural low-level features into mid-level features, still in an unimodal way, to build shots' representations. Figure 2 illustrates this process, taking the visual case as an example. The first step is to allocate visual or aural low-level features to a common space for further processing. This space is called Bag-of-Features (BoF, in Figure 2).

The second step is clustering similar feature vectors (Clusters, in Figure2). In this work, we have used the k-means clustering algorithm, recognized as a simple and fast method for numeric data. We used the default accuracy and maximum number of iterations provided by the
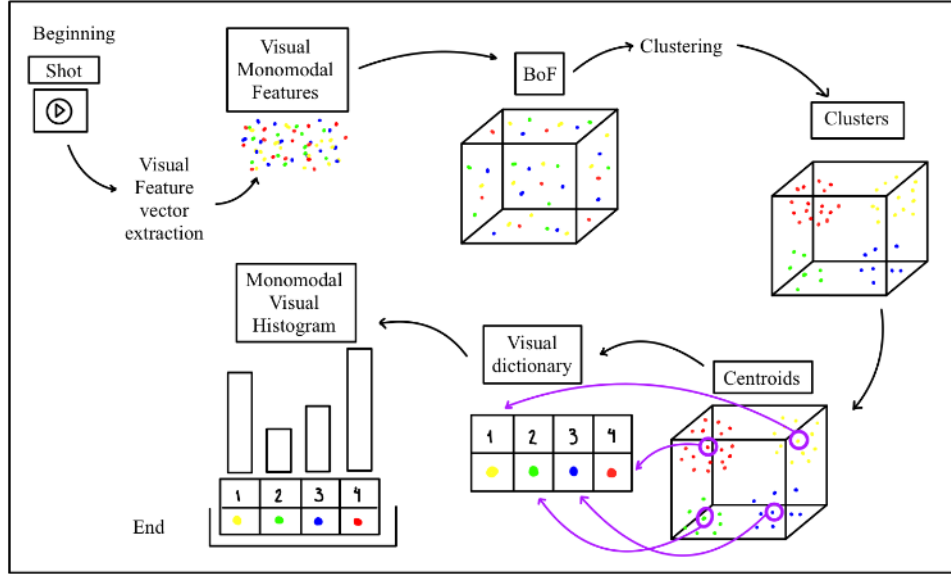
Figure 2. Mid-level monomodal features representation. Taking the visual case as an example. The same rationale applies to other modalities.

OpenCV implementation ($TERM\_CRITERIA\_EPS + TERM\_CRITERIA\_MAX\_ITER$). The optimum number of clusters (the k parameter) was 100 for the used video dataset (Section 5.3), and it was defined using the silhouette analysis method (Shutaywi and Kachouie [2021]). The next step establishes a dictionary of visual/aural words, also known as Bag of Visual/Aural Words (BoVW/BoAW). This is done by selecting all clusters' centroids (Centroids in Figure 2) and building a k-dimensional feature vector - the Visual/Aural Dictionary ($Visual dictionary$ in Figure 2).

Finally, a shot representation is built as a monomodal visual/aural histogram ($Monomodal Visual Histogram$ in Figure 2). This process is done by building a k-dimensional feature vector (the histogram) where each bin counts how many times each correspondent visual/aural word appears in a given shot. This count is done by comparing the (low-level) feature vectors from the shot with the dictionary words through a similarity measurement - the cosine similarity measure was chosen since it is adequate to measure semantic similarity between vectors. The measure that results in the smallest distance will indicate the visual/aural word corresponding to that feature vector. In the histogram, the frequency of this visual word bin will be increased by 1.

The rationale behind this representation is that shots with similar histograms exhibit comparable patterns within a given modality. In this way, similarity comparisons between histograms of the same modality applies. However, although different histograms representing different modalities have the same dimensions and comparable pattern counting, the patterns represented by each bin are distinct and uncorrelated. Therefore, direct similarity comparisons between histograms from diverse modalities have no meaning. Hence, a fusion operation is necessary to transfer the unimodal information from heterogeneous spaces into a unified multimodal one.

4.3 Feature Fusion

The feature fusion process consists of inputting two mid-level representations of a shot, one visual and another aural (unimodal histograms, as detailed in Sub-section 4.2), and applying a fusion operator. The output generated by an operator is a new histogram, with the same dimensionality as the inputs (k-dimensional), containing information from visual and aural modalities - a new multimodal shot representation. The fusion operators of interest in this work are those defined in Section 3.

4.4 Scene Segmentation Algorithm

We have used the baseline Scene Transition Graph (STG) algorithm Yeung *et al.* [1998] for segmenting a video into scenes, as defined by Kishi *et al.* [2019]. STG is suitable for our purposes, as the primary objective of this work is not to achieve high segmentation accuracy but rather to analyze the accuracy behavior of different fusion operators when applied to a temporal video segmentation task. In the experiments (see Sub-section 5.3), the pipeline was configured with fixed modules—including the segmentation—with the exception of the fusion module. The fusion operator was varied in each experimental run, and the segmentation efficacy was measured accordingly. This approach allows for an indirect evaluation of the operators' efficacy. Consequently, any segmentation algorithm can serve as the pipeline's segmentation module. We selected STG because it is simple, well-known, and a common baseline algorithm

The algorithm takes the multimodal fused histograms as input. In this approach, video shots are grouped using the hierarchical clustering algorithm with complete linkage, adapted to the scene segmentation task to avoid including temporally distant shots despite their similarity. The distance calculation, $\hat{d}max$, is given by:

$$\hat{d}max(C_i, C_j) = \max_{x \in C_i, y \in C_j} \hat{d}(x, y) \qquad (7)$$

where C_i and C_j are the shot clusters. $\hat{d}(S_i, S_j)$ is determined by $d(S_i, S_j)$ if $d_t(S_i, S_j) \leq T$, or ∞ otherwise.

Here, d represents the adopted distance measure in the algorithm, d_t is the temporal distance between shots, and T is a temporal window. The stopping criterion is defined by a parameter δ. Grouping stops when $\hat{d}_{max}(A,B) > \delta$ for all pairs of groups (A,B) with $A \neq B$. The resulting groups correspond to the vertices in the scene transition graph. An edge is created between vertices A and B if A contains a shot temporally adjacent to any shot in B.

The cut edges in the graph identify scene transitions, with each subgraph representing a scene. In the experiments, cosine distance was used with parameters T and δ set to 7 and 0.35, respectively. Those values were obtained empirically by a silhouette analysis on the video dataset.

5 Experiments

This section provides a detailed account of the experiments conducted during the study. Subsection 5.1 outlines the commonly used metrics in this field, elaborates on their distinctions, and offers a rationale for the selected metrics in this work. Subsection 5.2 offers an overview of the widely recognized BBC dataset. Finally, Subsection 5.3 presents an in-depth discussion of the experimental results.

5.1 Metrics

In the field of Multimedia Analysis, commonly used metrics for evaluating video tasks include Precision (P), Recall (R), and the F$_1$-score (F_1), with the latter being the harmonic mean of the former two Smeulders *et al.* [2000]. Although these metrics effectively measure efficacy and are widely adopted in the Multimedia area, they fall short in the TVSS context due to their inflexibility. Specifically, they treat the misclassification of one shot in a scene with the same severity as the misclassification of two, ten, or more shots in the same scene. This behavior does not account for the varying degrees of impact as the number of misclassified shots increases. To address these limitations, Vendrig and Worring [2002] proposed the metrics Coverage (C) and Overflow (O), which were reformulated by Han and Wu [2011]. Coverage measures the number of correctly predicted transition shots within the ground-truth scene boundaries, while Overflow quantifies the number of transition shots predicted beyond those boundaries. These metrics are more suitable for the TVSS task, as they better capture the nuanced differences in misclassification severity compared to the exact hits and misses measured by P and R. The formulations for C and O according to Han and Wu [2011] are as follows, respectively, in Equations 8 and 9:

$$C_t = \frac{\max_{i=1}^{|S|} |s_i \cap \bar{s}_t|}{|\bar{s}_t|} \tag{8}$$

$$O_t = 1 - \frac{|s_t|}{\sum_{i=1}^{|S|} |\bar{s}_t| \times \min(1, |s_i \cap \bar{s}_t|)} \tag{9}$$

where $S = \{s_1, \ldots, s_{|S|}\}$ is the set of predicted scenes, $\bar{S} = \{\bar{s}_1, \ldots, \bar{s}_{|\bar{S}|}\}$ is the set of real scenes and $|s_i|$ is the number of shots in a scene s_i. C_t of a scene $\bar{s}_t$ measures the number of shots of $\bar{s}_t$ correctly grouped together in the prediction, which corresponds to the longest overlap between s_i and $\bar{s}_t$. O_t of a scene t measures how many shots of t aren't covered by the prediction. C and O of an entire video are given by Equations 10 and 11:

$$C = \sum_{t=1}^{|\bar{S}|} C_t \times \frac{|\bar{s}_t|}{\sum |\bar{s}_i|} \tag{10}$$

$$O = \sum_{t=1}^{|\bar{S}|} O_t \times \frac{|\bar{s}_t|}{\sum |\bar{s}_i|} \tag{11}$$

The F_1-score between C and O is calculated by the harmonic mean of C and $1 - O$ since lower overflow values represent better segmentation.

In this work, we have adopted C and O metrics. However, we also present results (Subsection 5.3) using the F$_1$-score for P and R in order to allow comparisons with some of the related works.

5.2 Video Dataset

The BBC Dataset is a well-known dataset for temporal video segmentation introduced by Baraldi *et al.* [2015], comprising 11 videos from the BBC Planet Earth documentary series [3] about varied natural habitats on planet Earth. Each video is approximately 50 minutes long, coded in MP4 format (H.264/AVC video codec and MPEG-4 AAC audio codec) with 360x288 pixels, 25 FPS, and stereo channels with a sample rate of 48 kHz. This dataset includes ground truth annotations, listing 4916 shots and 672 scenes.

5.3 Results

This section presents the results of evaluating the efficacy of logical fusion operators using the TVSS pipeline described in Section 4. The goal is to determine the effectiveness of logical fusion operators compared to arithmetic fusion operators. Arithmetic fusion operators are a viable alternative in the TVSS task domain (Beserra and Goularte [2023]). However, logical fusion operators, despite being more efficient in terms of computational time (Patterson and Hennessy [2021]), remain underinvestigated.

An experiment was conducted by setting up the pipeline described in Section 4 (Figure 1) with a round-based fusion operator. We compared arithmetic operators (Sum, Max, and Average (Avg)) with logical operators (And, Or, and Xor). For each round, a multimodal fused representation was generated (Section 4.3) using one of the fusion operators, for each shot in each of the 11 videos of the BCC Dataset. This set of multimodal representations for each video's shot was then provided as input to the STG algorithm (Section 4.4). The segmentation results were compared with the dataset's Ground Truth by calculating the efficacy measures Coverage (C), Overflow (O), and F$_1$-score (F$_{1_{co}}$) (Section 5.1). The only modification in the TVSS pipeline, for each round, is a replacement of the fusion operator. In this way, a comparison of the segmentation results, using each of the operators, can give an evaluation of the operators efficacy. In total, 8 rounds were conducted: one for each of the 6 fusion operators and two additional rounds using unimodal representations (visual only and aural only). The purpose of comparing

---

[3] https://www.bbc.co.uk/programmes/b006mywy/episodes/guide

with unimodal representations is to confirm the advantages of multimodal fusion.

Table 1 presents the obtained results. The measures (C, P and $F_{1_{co}}$) are presented by each multimodal fusion operator (columns Sum to Xor) plus the results from each monomodal segmentation (columns $Visual$ and $Aural$). The left-most column ($Video/Episode$) identifies the BCC Dataset videos. At the bottom, Table 1 presents the metric average ($Average$) and the F_1-score standard deviation ($F_{1_{co}}$ $Std.Dev.$).

Upon analyzing the metrics' values from Table 1 across the dataset episodes, an expected pattern emerges: the metrics' values generally vary for each line. However, an exception is observed in the episode entitled $Caves$, which consistently presents identical values in multiple instances. This anomaly can be attributed to the unique characteristics of this episode. Unlike the others, the video shots in $Caves$ are predominantly captured in open, dark environments, featuring numerous small flying or floating objects (such as birds, bats, and fish) in the foreground, or in close-up shots of rock walls and cave entrances. Additionally, the audio in this episode only occupies a small portion of each shot and does not continue seamlessly from one shot to another. Consequently, SIFT features fail to provide accurate image representations, while MFCC is unable to offer sufficient audio data to distinctly represent each shot.

Now analyzing the aural and visual F_1-score results, aural monomodal segmentation consistently outperforms visual monomodal one. This can be attributed to two key factors: 1) the aural stream (documentary narrator) provides a coherent narrative with consistent semantics, unlike the visual stream, where images and video sequences change rapidly; 2) local features (SIFT) from the visual stream focus on small pieces of content, making it harder to find correlations between video sequences within a shot. Different choices for visual features may improve modalities' complementarity and final results. Nevertheless, multimodal approaches surpass monomodal ones, confirming that multimodal approaches improve results and demonstrate the complementarity of BCC Dataset modalities.

Observing the logical operators' results, the And operator is less effective, consistently showing lower $F_{1_{co}}$ values (13%) compared to Or and Xor (60%). This reflects the nature of the And operation, which negates information complementarity by turning a [1,0] input into a 0 output. Comparing Xor and Or operators, Xor is more stable. Both operators perform best in 3 out of 11 instances and have the same $F_{1_{co}}$ average, but Xor has a better O average (62% vs. 59%) and lower standard deviation (2 vs. 3). This indicates Xor is more stable when information complementarity exists.

Making a comparison between arithmetic and logical operators, the arithmetic Sum operator achieves the best overall performance with an average $F_{1_{co}}$ of 61%, outperforming in 4 out of 11 episodes. It is closely followed by Or and Xor, each with an average F_1-score of 60% and lower standard deviations.

Also, the execution time for both mathematical and logical operators was measured. The procedure involved calculating the average execution time over 100 runs for each of the six operators. Each run measured the time to compute a fusion operation over a pair of monomodal histograms randomly selected. As the histograms are 100-dimensional, the fusion operation performed 100 logical/arithmetic operations. Therefore, the resulting execution time value was divided by the dimensionality (100), giving an approximate average execution time for a single logical/arithmetic operation.

The results, in seconds, were Sum = 0.00053715705871582, Avg = 0.00109481811523438, Max = 0.0106735229492188, And = 0.0002305114746, Or = 0.0001430511475, and Xor = 0.0001525573734. These results demonstrate that both the logical operators Or and Xor are competitive alternatives to the arithmetic ones. They are almost as effective as the arithmetic operators, although more efficient.

We compared arithmetic and logical operators with three recent state-of-the-art TVSS methods: TransNet Soucek and Lokoc [2024], VSMBD Tan *et al.* [2024], and Dual Xing *et al.* [2024], all using the BBC Dataset. These related works report their experimental results using the F_1-score ($F_{1_{pr}}$) (Subsection 5.1) and omit Precision (P) and Recall (R) values. Table 2 presents the comparison results, showing the average $F_{1_{pr}}$ for each operator and the related works. By following the same pipeline procedure described earlier, we now calculated the average $F_{1_{pr}}$. The Sum arithmetic operator outperformed the state-of-the-art methods with an $F_{1_{pr}}$ of 57%, followed by the logical Or and Xor operators, each with an $F_{1_{pr}}$ of 52%. Although Sum achieved 5% higher $F_{1_{pr}}$ than the Dual and VSMBD methods (and 4% more than Xor and Or), it is important to note that $F_{1_{pr}}$ is rigid and heavily penalizes over- and under-segmentations. When using the more suitable TVSS metrics Coverage (C) and Overflow (O), the noticed differences tend to be smaller.

6 Conclusions

This work investigates the application of logical fusion operators (And, Or, Xor) at the mid-level feature space for the TVSS task. The proposed use of these operators for multimodal fusion has several advantages: no restrictions on the number of input features or types of modalities; effective reduction of data volume; applicability in different video analysis tasks as long as using feature vectors as information representation; simple and explainable methodology; fusion operators may be used as baselines for more complex methods, based on neural network models or not.

A comparative analysis (Section 5) using the public BBC Dataset demonstrates that Or and Xor logical operators are viable alternatives to arithmetic ones, with only a 1% lower efficacy on average and reduced computational time. This is especially important in scenarios where computational time is more critical than minor efficacy differences. Even when compared to neural networks state-of-the-art methods, fusion operators remain competitive. The Sum arithmetic operator outperforms them by an average of 5% in efficacy, followed by And and Xor operators with a margin of 1%. This suggests the compared neural models have the fusion mechanism highly coupled with the task or the correlation extraction module, introducing noise and diminishing

Table 1. Metrics measures obtained from BBC Dataset: coverage (C), overflow (O), F_1-score ($F_{1_{co}}$), and their averages ($Average$), in percentage (%), and standard deviation ($F_{1_{co}}$ $Std.Dev$). Highlighted values (*X*) mean the best F_1-score for that row (video) and bold font values (**X**) mean the second best. Higher values are better.

Video / Episode	Operators (%)							
	Sum	Max	Avg	And	Or	Xor	Visual	Aural
	C, O, $F_{1_{co}}$	C, O, $F_{1_{co}}$	C, O, $F_{1_{co}}$	C, O, $F_{1_{co}}$	C, O, $F_{1_{co}}$	C, O, $F_{1_{co}}$	C, O, $F_{1_{co}}$	C, O, $F_{1_{co}}$
From Pole to Pole	66, 57, 61	64, 57, 60	66, 55, 60	98, 03, 06	65, 59, 61	66, 58, **62**	100, 03, 06	55, 73, *63*
Mountains	64, 62, **63**	64, 60, 62	64, 62, **63**	98, 04, 08	65, 57, 60	65, 64, *64*	99, 04, 07	40, 80, 58
Ice Worlds	68, 61, **64**	65, 63, **64**	68, 61, **64**	100, 03, 05	66, 60, 63	61, 69, *65*	100, 03, 05	53, 77, 63
Great Plains	76, 62, **68**	78, 61, **68**	76, 62, **68**	98, 05, 14	78, 62, *69*	67, 69, **68**	84, 33, 47	64, 75, *69*
Jungles	66, 64, 65	69, 58, 63	66, 64, 65	79, 29, 42	67, 62, **65**	63, 63, 63	96, 17, 29	62, 73, *67*
Seasonal Forests	71, 59, *64*	68, 52, 59	70, 59, *64*	92, 12, 22	67, 57, 61	67, 56, 61	86, 24, 37	59, 68, **63**
Fresh Water	62, 75, **67**	62, 67, 66	61, 75, **67**	93, 10, 17	65, 71, *68*	59, 74, 65	85, 31, 46	47, 85, 60
Ocean Deep	63, 73, *68*	61, 75, 67	63, 72, 67	100, 04, 08	59, 75, 66	58, 77, 66	97, 08, 14	56, 75, 64
Shallow Seas	69, 71, **70**	69, 70, **70**	68, 70, 69	99, 03, 06	71, 71, *71*	65, 75, **70**	76, 59, 67	51, 80, 62
Caves	99, 04, *08*	99, 04, *08*	99, 04, *08*	100, 03, **06**	100, 03, **06**	99, 04, *08*	100, 03, **06**	99, 04, *08*
Deserts	65, 69, *67*	58, 71, 64	65, 69, *67*	97, 05, 09	60, 72, 66	53, 77, 63	99, 04, 09	55, 82, 66
Average	70, 60, *61*	68, 58, 59	70, 59, **60**	95, 05, 13	69, 59, **60**	66, 62, **60**	93, 17, 25	59, 70, 58
$F_{1_{co}}$ **Std. Dev.**	4	2	2	2	3	2	2	4

Table 2. BCC Dataset F_1-score ($F_{1_{pr}}$) averages from each fusion operator compared with F_1-score from Related Work. Highlighted (*x*) values mean the best $F_{1_{pr}}$ score and bold font values (**x**) mean the second best. Higher values are better.

	Operators (%)						Related Work (%)		
	Sum	Max	Avg	And	Or	Xor	Dual	VSMBD	TransNet
	$F_{1_{pr}}$	$F_{1_{pr}}$	$F_{1_{pr}}$	$F_{1_{pr}}$	$F_{1_{pr}}$	$F_{1_{pr}}$	$F_{1_{pr}}$	$F_{1_{pr}}$	$F_{1_{pr}}$
Average	*57*	52	**53**	18	**53**	**53**	52	52	50

the fusion efficacy. This was also pointed out by Beserra *et al*. [2020].

As limitations, this study used only one dataset and a local-based visual feature extraction. Future work should include datasets from different video domains and employ global or semantic visual feature extraction aiming to enhance both the generalization of fusion operators and task efficacy. Expanding this approach to other tasks also presents an interesting direction for future research.

Declarations

Funding

Authors' Contributions

Letícia B. Barbosa contributed to the study with conceptualization, data curation, formal analysis, investigation, software, validation, and writing – original draft. Rudinei Goularte contributed to this study with conceptualization, formal analysis, funding acquisition, methodology, project administration, supervision, and writing – review & editing. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

The datasets, including software (source code), the Ground-Truth files as proposed by Baraldi and Cucchiara (2015), and link for the videos files, generated and/or analysed during the current study are available at `https://github.com/rudineigoularte/EvaluatingLogicalFusionOperators`.

References

Ashutosh, K., Xue, Z., Nagarajan, T., and Grauman, K. (2024). Detours for Navigating Instructional Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18804–18815. DOI: 10.1109/CVPR52733.2024.01779.

Baraldi, L., Grana, C., and Cucchiara, R. (2015). A deep siamese network for scene detection in broadcast videos. In *Proceedings of the 23rd ACM International Conference on Multimedia*, pages 1199–1202. DOI: 10.1145/2733373.2806316.

Beserra, A. A. R. and Goularte, R. (2023). Multimodal early fusion operators for temporal video scene segmentation tasks. *Multimed Tools and Applications*, 82:31539–31556. DOI: 10.1007/s11042-023-14953-6.

Beserra, A. A. R., Kishi, R. M., and Goularte, R. (2020). Evaluating early fusion operators at mid-level feature space. In *Proceedings of the Brazilian Symposium on Multimedia and the Web*, pages 113–120. DOI: 10.1145/3428658.3431079.

Gross, B. M. (1965). The Managing of Organizations: The Administrative Struggle. *The ANNALS of the American Academy of Political and Social Science*, 1-2(1):197–198. DOI: 10.1177/000271626536000140.

Gôlo, M. P. S., de Moraes Junior, M. I., Goularte, R., and Marcacini, R. M. (2024). Unsupervised Heterogeneous Graph Neural Networks for One-Class Tasks: Exploring Early Fusion Operators. *Journal on Interactive Systems*, 15(1):517–529. DOI: 10.5753/jis.2024.4109.

Han, B. and Wu, W. (2011). Video scene segmentation using a novel boundary evaluation criterion and dynamic programming. In *2011 IEEE International*

Conference on Multimedia and Expo, pages 1–6. DOI: 10.1109/ICME.2011.6012001.

Jangra, A., Mukherjee, S., Jatowt, A., Saha, S., and Hasanuzzaman, M. (2023). A Survey on Multi-modal Summarization. *ACM Comput. Surv.*, 55(13s). DOI: 10.1145/3584700.

Jhuo, I.-H., Ye, G., Gao, S., Liu, D., Jiang, Y.-G., Lee, D., and Chang, S.-F. (2014). Discovering joint audio–visual codewords for video event detection. *Machine Vision and Applications*, 25:33–47. DOI: 10.1007/s00138-013-0567-0.

Jia, H. and Lao, H. (2022). Deep learning and multimodal feature fusion for the aided diagnosis of Alzheimer's disease. *Neural Comput. Appl.*, 34(22):19585–19598. DOI: 10.1007/s00521-022-07501-0.

Kishi, R. M., Trojahn, T. H., and Goularte, R. (2019). Correlation based feature fusion for the temporal video scene segmentation task. *Multimedia Tools and Applications*, 78:15623–15646. DOI: 10.1007/s11042-018-6959-4.

Koprinska, I. and Carrato, S. (2001). Temporal video segmentation: A survey. *Signal Processing: Image Communication*, 16(5):477–500. DOI: 10.1016/S0923-5965(00)00011-4.

Liu, Z., Cheng, J., Liu, L., Ren, Z., Zhang, Q., and Song, C. (2022). Dual-stream cross-modality fusion transformer for RGB-D action recognition. *Knowledge-Based Systems*, 255:109741. DOI: 10.1016/j.knosys.2022.109741.

Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110. DOI: 10.1023/B:VISI.0000029664.99615.94.

Patterson, D. A. and Hennessy, J. L. (2021). *Computer Organization and Design: The Hardware/Software Interface*. Morgan Kaufmann, Waltham, 5th edition.

Pereira, L. M., Salazar, A., and Vergara, L. (2024). A comparative study on recent automatic data fusion methods. *Computers*, 13(1). DOI: 10.3390/computers13010013.

Samadiani, N., Huang, G., Luo, W., Chi, C.-H., Shu, Y., Wang, R., and Kocaturk, T. (2022). A multiple feature fusion framework for video emotion recognition in the wild. *Concurrency and Computation: Practice and Experience*, 34(8):e5764. DOI: 10.1002/cpe.5764.

Sen, S., Dutta, A., and Dey, N. (2019). *Audio processing and speech recognition*. Springer, Berlin, 1st edition.

Shutaywi, M. and Kachouie, N. N. (2021). Silhouette analysis for performance evaluation in machine learning with applications to clustering. *Entropy*, 23(6). DOI: 10.3390/e23060759.

Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., and Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380. DOI: 10.1109/34.895972.

Snoek, C. G. M., Worring, M., and Smeulders, A. W. M. (2005). Early versus late fusion in semantic video analysis. In *Proceedings of the 13th Annual ACM International Conference on Multimedia*, MULTIMEDIA '05, page 399–402, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/1101149.1101236.

Soucek, T. and Lokoc, J. (2024). TransNet V2: An Effective Deep Network Architecture for Fast Shot Transition Detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, MM '24, page 11218–11221, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3664647.3685517.

Tan, J., Yang, P., Chen, L., and Wang, H. (2024). Temporal Scene Montage for Self-Supervised Video Scene Boundary Detection. *ACM Trans. Multimedia Comput. Commun. Appl.*, 20(7). DOI: 10.1145/3654669.

Vendrig, J. and Worring, M. (2002). Systematic evaluation of logical story unit segmentation. *IEEE Transactions on Multimedia*, 4(4):492–499. DOI: 10.1109/TMM.2002.802021.

Wei, C., Chen, Y., Chen, H., Hu, H., Zhang, G., Fu, J., Ritter, A., and Chen, W. (2024). UniIR: Training andnbsp;Benchmarking Universal Multimodal Information Retrievers. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LXXXVII*, page 387–404, Berlin, Heidelberg. Springer-Verlag. DOI: 10.1007/978-3-031-73021-4_23.

Xing, L., Tran, Q., Caba, F., Dernoncourt, F., Yoon, S., Wang, Z., Bui, T., and Carenini, G. (2024). Multi-modal video topic segmentation with dual-contrastive domain adaptation. In *MultiMedia Modeling*, pages 410–424, Cham. Springer Nature Switzerland. DOI: 10.1007/978-3-031-53311-2_30.

Yang, Z., Xu, L., Zhao, L., and Sharma, K. (2022). Multimodal Feature Fusion Based Hypergraph Learning Model. *Intell. Neuroscience*, 2022. DOI: 10.1155/2022/9073652.

Yeung, M., Yeo, B.-L., and Liu, B. (1998). Segmentation of video by clustering and graph analysis. *Computer Vision and Image Understanding*, 71(1):94–109. DOI: 10.1006/cviu.1997.0628.