P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

# On the use of audio events for improving video scene segmentation

Panagiotis Sidiropoulos[1,2], Vasileios Mezaris[1], Ioannis Kompatsiaris[1], Hugo Meinedo[3], Miguel Bugalho[3,4], and Isabel Trancoso[3,4]

[1] Information Technologies Institute / Centre for Research and Technology Hellas, 6th Km Charilaou-Thermi Road, Thermi 57001, Greece
[2] Center for Vision, Speech and Signal Processing / Faculty of Engineering and Physical Sciences, University of Surrey, Guildford, Surrey GU2 5XH, UK
[3] INESC-ID Lisboa, Portugal
[4] IST/UTL, Rua Alves Redol 9, 1000-029 Lisboa, Portugal
{psid, bmezaris, ikom}@iti.gr, {hugo.meinedo, mmfb}@l2f.inesc-id.pt, Isabel.Trancoso@inesc-id.pt

**Abstract.** This work deals with the problem of automatic temporal segmentation of a video into elementary semantic units known as scenes. Its novelty lies in the use of high-level audio information, in the form of audio events, for the improvement of scene segmentation performance. More specifically, the proposed technique is built upon a recently proposed audio-visual scene segmentation approach that involves the construction of multiple scene transition graphs (STGs) that separately exploit information coming from different modalities. In the extension of the latter approach presented in this work, audio event detection results are introduced to the definition of an audio-based scene transition graph, while a visual-based scene transition graph is also defined independently. The results of these two types of STGs are subsequently combined. The results of the application of the proposed technique to broadcast videos demonstrate the usefulness of audio events for scene segmentation and highlight the importance of introducing additional high-level information to the scene segmentation algorithms.

**Keywords:** Video analysis, scene segmentation, audio events, scene transition graph

## 1 Introduction

Video temporal decomposition into elementary semantic units is an essential pre-processing task for a wide range of video manipulation applications, such as video indexing, non-linear browsing, classification, etc. Video decomposition techniques aim to partition a video sequence into segments, such as shots and scenes, according to semantic or structural criteria. Shots are elementary structural segments that are defined as sequences of images taken without interruption by a single camera [1]. On the other hand, scenes are often defined as Logical Story Units (LSU) [2], i.e., as a series of temporally contiguous shots characterized by overlapping links that connect shots with similar content. Figure 1 illustrates the relations between different kinds of temporal segments of a video.

Early approaches to scene segmentation focused on exploiting visual-only similarity among shots [2, 3], to group them into scenes. In [3], the Scene Transition Graph (STG) was originally presented. The Scene Transition Graph method exploits the visual similarity between key-frames of video shots to construct a connected graph, whose cut-edges constitute the set of scene boundaries. Another recent uni-modal scene segmentation technique [4] uses spectral clustering to conduct shot grouping, without taking into account temporal proximity. Subsequently, the clustering outcome is used for assigning class labels to the shots, and the similarity between label sequences is used for identifying the scene boundaries.

In the last years, several scene segmentation methods that exploit both the visual and auditory channel have been developed, including [5–8]. In [5], a fuzzy k-means algorithm is used for segmenting the auditory channel of a video into audio segments, each belonging to one of 5 classes (silence, speech, music etc.). Following the assumption that a scene change is associated with simultaneous change of visual and audio characteristics, scene breaks are identified when a visual shot boundary exists within an empirically-set time interval before or after an audio segment boundary. In [6], visual information usage is limited to the stage of video shot segmentation. Subsequently, several low-level audio descriptors (i.e., volume, sub-band energy, spectral and cepstral flux) are extracted for each shot. Finally, neighboring shots whose Euclidean distance in the low-level audio descriptor space exceeds a dynamic threshold are assigned to different scenes. In [7], audio and visual features are extracted

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

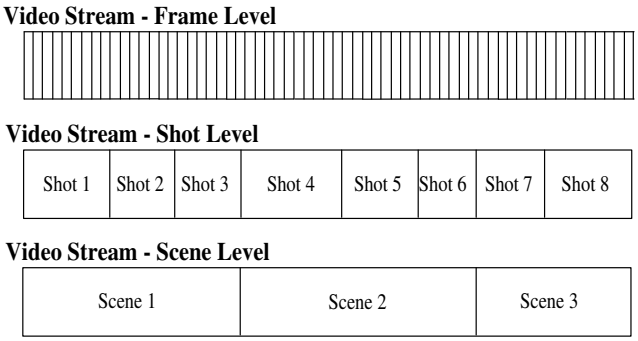


**Fig. 1.** Video stream decomposition to frames, shots and scenes.

for every visual shot and serve as input to a Support Vector Machines (SVM) classifier, which decides on the class membership (scene-change / non-scene-change) of every shot boundary. However, this requires the availability of sufficient training data. Although audio information has been shown in these and other previous works to be beneficial for the task of scene segmentation, higher-level audio features such as speaker clustering or audio event detection results are not frequently exploited. In a recent work [8], the use of audio scene changes and automatic speech recognition (ASR) transcripts together with visual features is proposed; audio scene changes are detected using a multi-scale Kullback-Leibler distance and low-level audio features, while latent semantic analysis (LSA) is used for calculating the similarity between temporal fragments of ASR transcripts. In [9], the combined use of visual features and some high-level audio cues (namely, speaker clustering and audio background characterization results) for constructing scene transition graphs was proposed.

In this work, this definition of the scene as a Logical Story Unit is adopted and the method of [9] is extended in order to exploit richer high-level audio information. To this end, a large number of audio event detectors is employed, and their detection scores are used for representing each temporal segment of the audio-visual medium in an audio event space. This representation together with an appropriate distance measure is used, in combination with previously exploited high-level audio (e.g. speaker clustering results) and low-level visual cues, for constructing a combination of different scene transition graphs (Multi-Evidence STG - MESTG) that identifies the scene boundaries. The rest of the chapter is organized as follows: an overview of the proposed approach is presented in Sect. 2. Audio event definition and the use of audio events in representing video temporal segments are discussed in Sects. 3 and 4, while Sect. 5 presents the proposed MESTG approach. Experimental results are presented in Sect. 6 and conclusions are drawn in Sect. 7.

## 2 Overview of the Proposed Approach

Scene segmentation is typically performed by clustering contiguous video shots; the proposed MESTG approach is no exception to this rule. Thus, scene segmentation starts with the application of the method of [1] for generating a decomposition $S$ of the video to visual shots,

$$S = \{s_i\}_{i=1}^{I} \ . \tag{1}$$

Subsequently, as illustrated in Fig. 2, visual feature extraction is performed. Audio segmentation, which includes, among others, speaker clustering and background classification stages [10] [11], is also performed in parallel. This audio segmentation process results in the definition of a partitioning of the audio stream,

$$\mathcal{A} = \{\alpha_x\}_{x=1}^{X} \ , \quad \alpha_x = [t_x^1, t_x^2] \ , \tag{2}$$

where $t_x^1$ and $t_x^2$ are the start- and end-times of audio segment $\alpha_x$. For each $\alpha_x$, the speaker identity of it and its background class are also identified during audio segmentation; we use $\sigma(\alpha_x)$ to denote the speaker identity of $\alpha_x$, if any, and $\beta(\alpha_x)$ to denote its background class. Audio event detection, as discussed in detail in the following section, is also performed. Using the resulting features, i.e.,

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

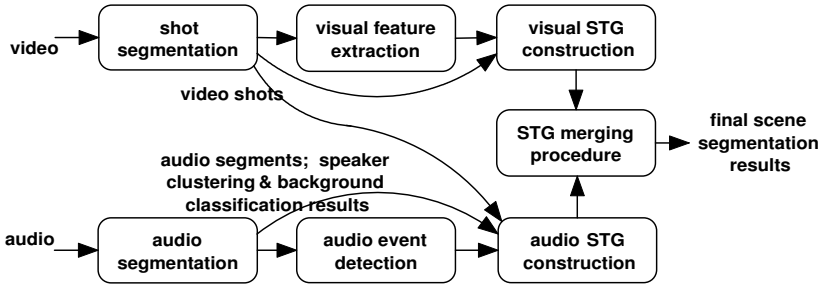


**Fig. 2.** Overview of the proposed scene segmentation scheme.

- HSV histograms of shot key-frames,
- Speaker clustering results,
- Audio background classification into one of three categories (noise, silence, music),
- Detection results (confidence values) for a multitude of audio events,

the proposed MESTG method proceeds with the definition of two types of scene transition graphs (audio STG, visual STG) and a procedure for subsequently merging their results.

## 3 Audio Events

For the purpose of scene segmentation, let us define an audio event as a semantically elementary piece of information that can be found in the audio stream of a video. Telephone ringing, dog barking, music, child voice, traffic noise, explosions are only at few of a wide range of possible audio events. As can be deduced from the audio event definition, more than one audio events may coexist in one temporal segment and may even temporally overlap with each other. For example, in a shot where a person stands by a street and talks, several speech- and traffic-related audio events are expected to coexist.

It is intuitionally expected that taking into account audio event detection results may contribute to improved video scene segmentation. This is based on the reasonable assumption that the presence of the same audio event in more than one adjacent or neighboring audio segments may be a good indication of their common scene membership. On the contrary, the presence of completely different audio events in adjacent temporal segments may be a good indication of their different scene membership, which reveals the presence of a scene boundary.

The first step in testing the validity of the above assumptions is the definition of a number of meaningful audio events and of appropriate methods for their detection. This work integrates two different sets of audio events. Different detection methodologies are used for each set.

### 3.1 Audio Segmentation

The first set includes the type of audio event that is dealt with by an audio segmentation (or diarization) module. Audio segmentation can mean many different things. In this chapter, we restrict its meaning to the type of segmentation that can be performed on the audio signal alone, without taking into account its linguistic contents. This type of segmentation can be done in several tasks. Acoustic Change Detection (ACD) is the task responsible for the detection of audio locations where speakers or background conditions have changed. Speech/Non-Speech (SNS) classification is responsible for determining if the audio contains speech or not (i.e., it results in a binary classification of the audio signal to either Speech or Non-Speech). Gender Detection (GD) distinguishes between male and female gender speakers (i.e., given a speech segment, it results in a binary classification of it to either Male or Female); however, an age-directed segmentation can be also useful as part of the gender detection task, for detecting children voices for instance. Background Conditions (BC) classification indicates whether the background audio signal (i.e, the audio signal, excluding any speech that may be part of it) is clean (:nothing is heard), musical (:music is heard), or noisy. Speaker Clustering (SC) identifies all the speech segments produced by the same speaker. Speaker Identification (SID) is the task of recognizing the identity of

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

**Table 1.** List of the 14 audio-segmentation related events.

| | | |
|---|---|---|
| Child Voice | Female Voice | Male Voice |
| Speech | Voice With Background Noise | Voice With Background Music |
| Music | Non Vocal Music | Vocal Music |
| Clean Background | Noise Background | Music Background |
| Telephone Band | People Talking | |

certain often recurring speakers, such as news anchors or very important personalities, by their voice, based on a classifier that is trained specifically for such speakers of interest (similarly in principle to how face recognition algorithms can be trained to identify specific people of interest, e.g. a particular political figure, by their faces). More recently, the term speaker diarization (SD) became synonymous to segmentation into speaker-homogeneous regions, answering the question "Who spoke when?". Altogether, 14 different events are automatically detected by this audio segmentation module (ex: Male Voice, Voice With Background Noise, Music, etc.) [12]. The list is included in Table 1. Note, however, that this figure does not include the information provided by the Speaker Clustering component on the cluster identity, which is also exploited for scene segmentation in this work.

The audio segmentation components are mostly model-based, making extensive use of feed-forward fully connected Multi-Layer Perceptrons (MLPs) that are trained with the back-propagation algorithm. All the classifiers (realizing tasks SNS, GD, BC, and SID, as defined above) share a similar architecture: a MLP with 9 input context frames of 26 coefficients (12th order Perceptual Linear Prediction (PLP) plus energy and deltas), two hidden layers with 250 sigmoidal units each and the appropriate number of softmax output units (one for each class), which can be viewed as giving a probabilistic estimate of the input frame belonging to that class. Despite the Acoustic Change Detection and Speech/Non-speech blocks being conceptually different, they were implemented simultaneously in the SNS component, considering that a speaker turn is most often preceeded by a small non-speech segment. The output of the SNS MLP classifier is smoothed using a median filter, and processed by a finite-state machine, involving confidence and duration thresholds. When a speaker change is detected, the first $t_{sum}$ frames of that segment are used to calculate gender, background conditions, and speaker identification classifications (e.g. anchors). Each classifier computes the decision with the highest average probability over all the $t_{sum}$ frames. The Speaker Clustering component, which uses an online leader-follower strategy, tries to group all segments uttered by the same speaker. The first $t_{sum}$ frames (at most) of a new segment are compared with all the same-gender clusters found so far. Two SC components are used in parallel (one for each gender). A new speech segment is merged with the cluster with the lowest distance, provided that it falls below a predefined threshold. The distance measure for merging clusters is a modified version of the Bayesian Information Criteria [11]. Our latest addition to the audio segmentation module is a telephone bandwidth detector. Given the lack of a large manually labeled corpus, a bootstrapping approach has been adopted in which a simple Linear Discriminant Analysis (LDA) classifier has been trained with a small amount of manually labeled data in order to generate automatic transcriptions for the posterior development of a binary MLP classifier. The adopted feature set consisted of 15 logarithmic filter bank energies extracted at a frame rate of 20 ms with a time shift of 10 ms, and corresponding deltas.

The background classifier was initially trained with only broadcast news data that had very limited examples of music and noisy backgrounds, and were inconsistently labeled in terms of these conditions. This motivated the development of alternative classifiers with extended training data reflecting a wide variety of conditions. The related detectors are: Music, Vocal Music, Non Vocal Music and Speech (another speech detector, using multi-layer perceptrons, also exists, corresponding to the People Talking event).

The new Gaussian mixture models (GMMs) included 1024 mixtures, and were trained using a different set of features (Brightness, Bandwidth, Zero Crossing Rate, Energy, Audio Spectrum Envelope and Audio Spectrum Centroid), extracted from 16 kHz audio, with 500 ms windows and 10 ms step. Silences were removed from the audio. Four models were trained: World, Speech, Non-Vocal Music, and Vocal Music.

Each of the GMM models was used to retrieve log likelihood values for each frame. Frame confidence values were calculated by dividing the log likelihood values for each model by the sum of all log likelihood values for all four models. The Vocal, Non-Vocal and Speech models were used for the Vocal Music, Non Vocal Music and Speech event detectors. The Music detector is the sum of the confidence values for the Vocal and Non-Vocal models. Segment confidence values were obtained by averaging the frame confidence values.

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

**Table 2.** List of 61 additional audio events corresponding to noise-like sounds.

| | | |
|---|---|---|
| Airplane Engine Jet | Airplane Engine Propeller | Animal Hiss |
| Baby Whining or Crying | Bear | Bell Electric |
| Bell Mechanic | Big Cat | Birds |
| Bite Chew Eat | Bus | Buzzer |
| Car | Cat Meowing | Chicken Clucking |
| Child Laughing | Cow | Crowd Applause |
| Digital Beep | Dog Barking | Dolphin |
| Donkey | Door Open or Close | Drink |
| Elephant or Trumpet | Electricity | Explosion |
| Fire | Fireworks | Frog |
| Glass | Gun Shot Heavy | Gun Shot Light |
| Hammering | Helicopter | Horn Vehicle |
| Horse Walking | Insect Buzz | Insect Chirp |
| Moose or Elk or Deer | Morse Code | Motorcycle |
| Paper | Pig | Rattlesnake |
| Saw Electric | Saw Manual | Sheep |
| Siren | Telephone Ringing Bell | Telephone Ringing Digital |
| Thunder | Traffic | Train |
| Typing | Walk or Run or Climb Stairs (Hard) | Walk or Run or Climb Stairs (Soft) |
| Water | Whistle | Wind |
| Wolf or Coyote or Dog Howling | | |

### 3.2 Finer Discrimination of Noisy Events

The second set of events targets a finer discrimination of noise-like sounds, such as Dog Barking, Siren, Crowd Applause, Explosion, etc. [13]. The greatest difficulty in building automatic detectors for this type of event is the lack of corpora manually labeled in terms of these events. This motivated the adoption of a very large sound effect corpus for training, given that it is intrinsically labeled, as each file typically contains a single type of sound. The corpus includes approximately 18,700 files with an estimated total duration of 289.6h, and was provided by one of the partners in the VIDIVIDEO project (B&G)[5]. The list of 61 events for which this corpus provided enough training material is shown in Table 2.

Most of the training files have a sampling rate of 44.1kHz. However, many were recorded with a low bandwidth ($<$ 10kHz). This motivated a uniform downsampling to 16 kHz. This corpus was used to train one-against-all detectors for each concept by building concept-specific and world models. Our initial set of detectors was SVM-based, and the experiments were made using the LIBSVM toolkit [14]. Preliminary experiments compared the performance of a limited set of features: Perceptual Linear Prediction (PLP) or Mel-Frequency Cepstral Coefficients (MFCC) coefficients (19 + energy + deltas), Zero Crossing Rate (ZCR), brightness, and bandwidth. The latter are, respectively, the first and second order statistics of the spectrogram, and they roughly measure the timbre quality. The world model was build using between 92 and 96 files, of which an average of 31 were used as the development set. As a starting point, analysis windows of 0.5s with 0.25s overlap were adopted. Three different kernels were considered for the SVM (linear, polynomial and radial basis function (RBF)). Overall, the best results were obtained with the latter kernel. The difference between the performance of MFCC and PLP coefficients was not significant.

As a result of the event detection process discussed in this and the previous section, a total of 75 audio events are defined and, based on the output of the corresponding detectors, a vector $EV$,

$$EV = [ev(1), ev(2), ..., ev(J)], \quad J = 75, \tag{3}$$

of confidence values is extracted and stored for each audio segment.

[5] Netherlands Institute for Sound & Vision, http://instituut.beeldengeluid.nl/

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

### 3.3 Audio Event Detection Performance

The Audio Segmentation components were tuned to the Broadcast New (BN) domain, which justifies the evaluation of their performance in a test set of six 1-hour long BN shows. The classification error rate of the SNS and GD blocks are comparable to the state of the art: 4.7% and 2.4%, respectively. [6] As explained, the BC results could not be considered reliable as the manual labels unfortunately lacked consistency.

The speaker clustering performance for news anchors shows very good results due to the SID models (4.1% Diarizition Error Rate (DER)). For the other speakers the results are not so good (26.0% DER). In part, these results can be attributed to the long duration of the BN shows, which have an average of 64 different speakers per news show, and also to the very large percentage of speech with loud background noise, mainly from street interviews.

The telephone bandwidth classifier was not evaluated in this data set, which did not include telephone data labels. The rate of correctly classified frames in the validation data set, obtained by the LDA classifier, was 99.8%. In other BN test sets, the rate achieved by the MLP was lower, which we also attributed to the high variability of the training data.

For the audio events of the second set, the performance was first evaluated in terms of F-measure, in a development set of sound effects. The results were generally very good (above 0.8). The worst results were obtained with Door, Fireworks, Hammering, and Saw Manual. The performance with real-life data (movies, documentaries, talk shows and broadcast news), however, is much more challenging than the classification of isolated events. The worse performance can often be due to the fact that audio events almost never occur separately, being corrupted by music, speech, background noise and/or other audio events.

## 4 Audio Event-based Segment Representation and Similarity Evaluation

For enabling the effective representation of temporal segments in the audio event space, and the evaluation of segment dissimilarity on the basis of audio events, two tasks are necessary: the normalization of the extracted audio event vectors, and the definition of an appropriate event vector distance measure.

Audio event vector normalization is motivated by the diversity of the distributions of confidence values among different event detectors for a given video. This is in part due to the differences in the actual frequency of appearance of different events within the video. For example, in a video with a female narrator speaking throughout the entire video and a thunder-like sound being heard in just a couple of shots, it is expected that the "female voice" audio event will receive very high confidence values in many shots, while the "thunder" audio event is likely to receive high or moderate confidence values in just the shots where the thunder-like sound is heard, and even lower values in all others. However, the high or moderate confidence values that the latter audio event receives should be considered as a strong indication in favor of those shots' common scene membership. In order for them to receive the due attention during scene segmentation, the normalization of confidence values depending on their distribution for each audio event is proposed, and a very simple (most likely non-optimal) normalization approach is adopted in this work. Specifically, if $ev(j)$ is the initial confidence value of the $j$-th audio event in a temporal segment, and $maxev_j$ is the maximum value of the $j$-th audio event in all the temporal segments of the video, then the normalized confidence value $\tilde{ev}(j)$ is:

$$\tilde{ev}(j) = \frac{ev(j)}{maxev_j} \ . \tag{4}$$

Following event vector normalization, the definition of a shot dissimilarity measure is based on the assumption that not only the difference of audio event confidence values between two segments, but also the absolute confidence values themselves, are important. Indeed, if for a given audio event two segments present similarly low confidence values, the only deduction that can be made is that this audio event is most probably not present in both segments; no conclusion can be drawn on the semantic similarity of these two segments. On the contrary, if two segments present similarly high confidence values, then it can be inferred that the same audio event is present in both segments, and this concurrence reveals a significant semantic similarity. The commonly used L1 distance or other Minkowski distances would not satisfy the above requirements, since they depend only on

[6] A recent version of the GD component achieved the first place in the Interspeech 2010 Paralinguistic Challenge in the category of Male/Female/Child classification [15].

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

the difference of the confidence values. Instead of them, a variation of the Chi-test distance is employed in this work. If $\tilde{EV}_1$, $\tilde{EV}_2$ are two normalized audio event vectors, then their distance $D$ is defined as:

$$D(\tilde{EV}_1, \tilde{EV}_2) = \sqrt{\sum_{j=1}^{J} \frac{(\tilde{ev}_1(j) - \tilde{ev}_2(j))^2}{\tilde{ev}_1(j) + \tilde{ev}_2(j)}} \ . \tag{5}$$

It can be seen that this dissimilarity measure does not depend only on the difference of the audio event vectors, satisfying the previously discussed dissimilarity measure requirements.

## 5 Multi-evidence Scene Transition Graph Method

### 5.1 Audio STG Definition

The definition of the ASTG is based on the following assumptions:

- Scene boundaries are a subset of the visual shot boundaries of the video (i.e., a visual shot cannot belong to more than one scenes).
- Each audio segment cannot belong to more than one scenes. The same holds for a set of temporally consecutive audio segments that share the same $\sigma(.)$, $\beta(.)$ values and exhibit similar audio events. Two audio segments are said to exhibit similar audio events if the distance between their audio event vectors, as defined in Sect. 4, is lower than an empirical threshold.
- Audio event similarity and the distribution of speaker identities across two shots (or two larger temporally contiguous video segments) can serve as measures of audio similarity.

Based on these assumptions, an ASTG is constructed as follows (Fig. 3):

- Step 1. The similarity of temporally adjacent audio segments $\alpha_x$, $\alpha_{x+1}$ is examined, starting from $\alpha_1$. Denoting $\tilde{EV}_x$, $\tilde{EV}_{x+1}$ the audio event vectors of $\alpha_x$, $\alpha_{x+1}$ respectively, the two audio segments are merged if $\sigma(\alpha_x) = \sigma(\alpha_{x+1})$, $\beta(\alpha_x) = \beta(\alpha_{x+1})$, and $D(\tilde{EV}_x, \tilde{EV}_{x+1}) < T_{ev}$, where $T_{ev}$ is an empirically defined threshold. For simplicity, the audio segments resulting from this merging step and used in the next step continue to be denoted $\alpha_x$.
- Step 2. Merging of visual shots is performed: for every $\alpha_x$, the visual shots that temporally overlap with it by at least $T_a$ msec are merged to a video unit.
- Step 3. The video units formed in step 2 are clustered according to the dissimilarity $\Delta(.)$ of their speaker identity distributions and the distance $D(.)$ of their audio event vectors. The two dissimilarity measures are linearly combined to produce a one-dimensional distance measure. Assignment of two video units to the same cluster requires both this distance measure and the temporal distance between them to be lower than certain thresholds.
- Step 4. A connected graph is formed, in which the nodes represent the clusters of video units and a directed edge is drawn from a node to another if there is a shot included the first node that immediately precedes any shot included in the second node [3], [9]. The collection of cut-edges, i.e., the edges that, if removed, result in two disconnected graphs, constitutes the set of estimated video scene boundaries.

It should be noted that the speaker identity distribution of a video unit is:

$$H_x = [h_1 \ \ h_2 \ ... \ h_G] \ , \tag{6}$$

where $G$ is the total number of speakers in the video, according to the speaker clustering results, and $h_g$, $g = 1, \ldots, G$, is defined as the time that speaker $g$ is active in the video unit divided by the total duration of the same video unit. The $L1$ metric is used as similarity function $\Delta(H_x, H_y)$.

### 5.2 Visual STG Definition

Similarly to ASTG, a scene transition graph based on visual information (VSTG) is defined. The VSTG comprises nodes, which contain a number of visually similar and temporally neighboring shots, and edges which represent the time evolution of the story. Visual similarity of shots is evaluated by calculating the Euclidean distance of HSV-histogram vectors of shot key-frames. More details on the visual scene transition graph can be found in [3].

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

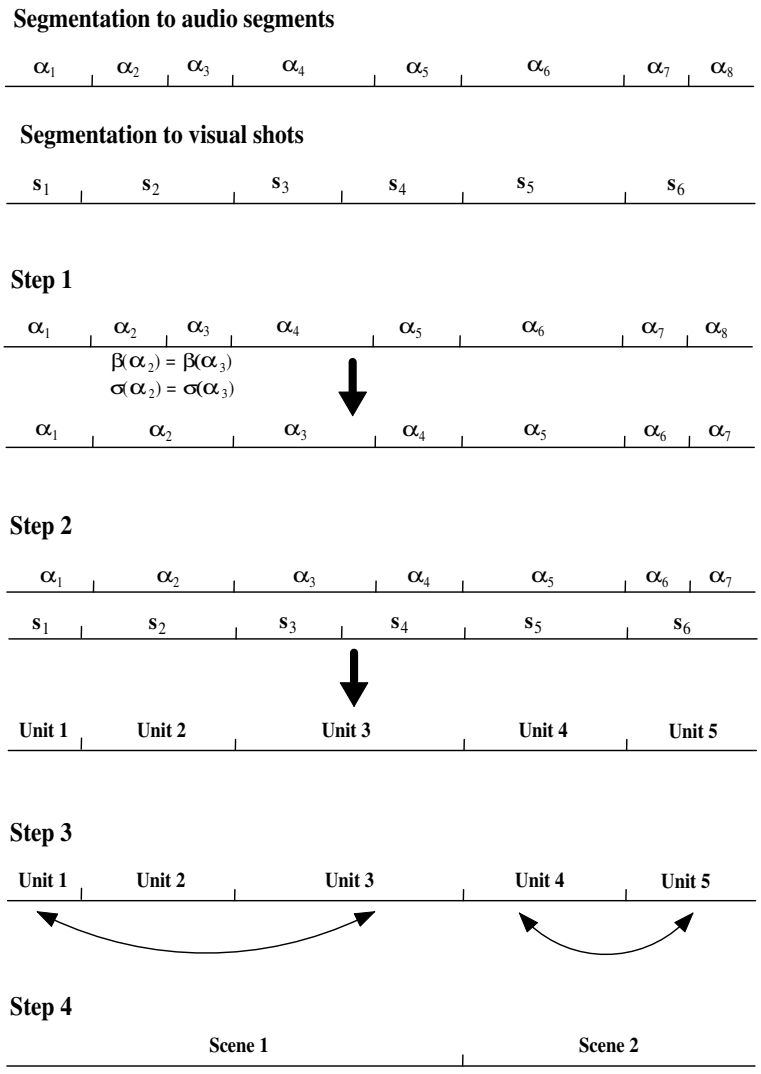


**Fig. 3.** An example of ASTG construction according to the algorithm of Sect. 5.1. The video stream is initially decomposed into 8 audio segments ($a_1$ to $a_8$) and 6 video shots ($s_1$ to $s_6$). Firstly, the audio segments that are adjacent and present same background class, speaker identity and also similar audio events are merged ($a_2$ and $a_3$). Subsequently, in step 2 shots $s_3$ and $s_4$, which overlap with audio segment $a_3$ by more than a threshold, are merged into a video unit. On the contrary, shots $s_1$ and $s_2$ are not merged, since the overlapping of $s_2$ with audio segment $a_1$ is minimal. In the third step, speaker identity distributions and audio event vectors are estimated for each video unit and their dissimilarity is used to determine which video units should be assigned to the same cluster (Unit 1 and Unit 3 are assigned to the same cluster; Unit 4 and Unit 5 are also assigned to a single cluster). Finally, the scene transition graph is constructed and as a result, in this example, the video units are joined to form 2 scenes.

### 5.3 Visual and Audio Scene Transition Graph Merging

In [9] we introduced a probabilistic scene transition graph merging approach that combines the visual and audio STGs and simultaneously reduces the dependency of the proposed approach on STG construction parameters. Similarly to this approach, in this work multiple VSTGs are created, each using a different randomly selected set of parameter values. Then, the fraction $p_i^v$ of VSTGs that identify the boundary between shots $s_i$ and $s_{i+1}$ as a scene boundary (i.e., the number of such VSTGs, divided by the total number of generated VSTGs) is calculated and used as a measure of our confidence on this being a scene boundary, based on visual information. The same procedure is followed for audio information using multiple ASTGs, resulting in confidence values $p_i^a$.

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

Subsequently, these confidence values are linearly combined to result in an audio-visual confidence value $p_i$:

$$p_i = V \cdot p_i^v + U \cdot p_i^a \ . \tag{7}$$

Finally, all shot boundaries for which $p_i$ exceeds a threshold form the set of scene boundaries estimated by the proposed MESTG approach. In the above formula, $U$ and $V$ are global parameters that control the relative weight of the ASTGs and VSTGs in the audio-visual scene boundary estimation.

## 6 Experimental Results

For experimentation, a test-set of 7 documentary films (229 minutes in total) from the collection of B&G was used. Application of the shot segmentation algorithm of [1] to this test-set and manual grouping of the shots to scenes resulted in 237 ground truth scenes. For evaluating the results of the proposed and other scene segmentation techniques, the Coverage and Overflow measures, proposed in [16] for scene segmentation evaluation, were employed. Coverage measures to what extent frames belonging to the same scene are correctly grouped together, while Overflow evaluates the quantity of frames that, although not belonging to the same scene, are erroneously grouped together. More detailed definitions of these two measures can be found in [16]. The optimal values for Coverage and Overflow are 100% and 0% respectively. The F-score is defined in this work as the harmonic mean of $C$ and $1 - O$, to combine Coverage and Overflow in a single measure,

$$F = \frac{2C(1-O)}{C + (1-O)} \ , \tag{8}$$

where $1 - O$ is used in the above definition instead of $O$ to account for 0 being the optimal value of the latter, instead of 1.

Using the above test-set and measures, the proposed approach (MESTG) was compared with the audio-visual scene segmentation technique (AVSTG) of [9], the methods of [5] and [4], and the visual scene transition graph (VSTG). For constructing the latter, the required parameter values were chosen by experimentation, as in [3]. For the MESTG and AVSTG approaches, the probabilistic merging procedure discussed in Sect. 5.3 was followed, involving the creation of 1000 ASTGs and 1000 VSTGs with different parameters for estimating the required probability values. Weights $V$, $U$ of (7) were tuned with the use of least squares estimation and one video manually segmented into scenes; the resulting values were 0.482 and 0.518 respectively. The results of experimentation are shown in Table 3, where it can be seen that the use of audio events in MESTG leads to an increase of Coverage by 1.89% and a decrease of Overflow by 0.34%, compared to the AVSTG. The MESTG approach also significantly outperforms the methods of [3], [5] and [4].

Furthermore, we have compared four different alternatives for constructing the audio scene transition graph. The first one (SP1) uses only the speaker identity distribution, while omitting steps 1 and 2 of the ASTG construction algorithm of Sect. 5.1. The other 3 variations use the proposed ASTG construction algorithm and differentiate only in terms of the considered audio descriptors. Specifically, SP2 makes use only of speaker identity distribution (6), whereas SPAE1 additionally employs the 14 audio events of Table 1. Finally, in SPAE2 the ASTG is built as proposed in this work, i.e, it exploits the speaker identity distribution, the 14 audio events of Table 1 and the 61 audio events of Table 2.

In the experimentation we examined the results of these variations both when they are used by themselves for scene segmentation and when each of them is combined with the visual STG, using the merging approach of Sect. 5.3. It should be noted that the combination of SP2 and the visual STG leads to the technique that is proposed in [9] (AVSTG), while the combination of SPAE2 and the visual STG results in the MESTG, presented in this work. The results of experimentation are shown in Table 4. It can be seen that none of the audio segmentation techniques can provide adequate scene segmentation accuracy when used in isolation. However, when combined with the visual STG, the additional improvement that each portion of the audio information contributes to can be seen by comparing the results of the last row of Table 4. Specifically, the proposed approach is shown to outperform the other 3 variations by at least 1.07% when used along with the VSTG. Finally, as it is shown in Table 4, omitting steps 1 and 2 of the ASTG construction algorithm reduces the system performance by 2.79%.

In Figs. 4 and 5 two examples of the outcome of MESTG, AVSTG and VSTG are shown. In contrary to MESTG, both the VSTG and AVSTG approaches fail to cluster all shots into a single scene in these examples.

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

**Table 3.** Performance evaluation of MESTG and comparison with literature works.

| Method | VSTG [3] | [5] | [4] | AVSTG [9] | MESTG |
|---|---|---|---|---|---|
| Coverage (%) | 79.18 | 77.93 | 70.13 | 83.86 | **85.75** |
| Overflow (%) | 17.81 | 13.88 | 21.93 | 11.05 | **10.71** |
| F-Score | 80.66 | 81.82 | 73.89 | 86.33 | **87.48** |

**Table 4.** Performance evaluation of 4 different audio STG variations in the documentary database. The first part of the table reports the performance of each variation when used by itself for scene segmentation. The second part reports the overall performance when each variation is combined with the visual STG as described in Sect. 5.3.

| Method | SP1 | SP2 | SPAE1 | SPAE2 |
|---|---|---|---|---|
| Coverage (%) | 58.7 | 67.14 | 58.86 | **69.29** |
| Overflow (%) | **20.32** | 26.67 | 28.77 | 31.72 |
| F-Score | 67.6 | **70.1** | 64.46 | 68.78 |
| Coverage (%) | 78.5 | 83.86 | 84.43 | **85.75** |
| Overflow (%) | 10.73 | 11.05 | 11.53 | **10.71** |
| F-Score | 83.54 | 86.33 | 86.41 | **87.48** |

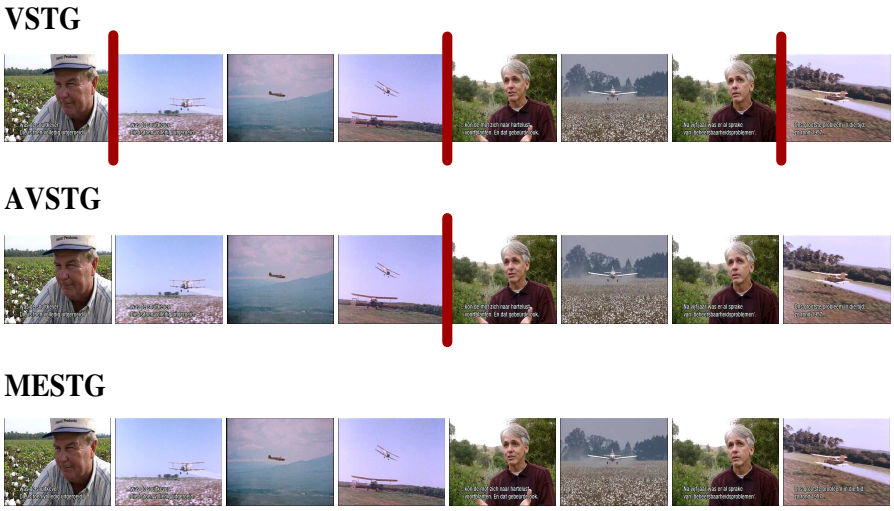


**Fig. 4.** A scene segmentation example. In each row, video shots are represented by one keyframe. According to the ground truth segmentation, all depicted shots belong to a single scene, related to field sprays in which shots of airplanes spraying interchange with farmers talking. It can be seen that the VSTG alone erroneously detects 3 scene boundaries, i.e., a scene boundary is declared in all shot boundary positions where the visual signal changes significantly, providing that there is no repetitive pattern (e.g. the same person re-appearing, as is the case with the second of the two farmers shown above). AVSTG cannot fully remedy this over-segmentation, whereas MESTG manages to assign all 8 shots to the same scene, making use of the common airplane sound that is found in all shots in which a speaker is not included.

## 7 Conclusions

In this work the use of high-level audio events for the improvement of scene segmentation performance was examined, and a multi-modal scene segmentation technique exploiting audio events and other audio-visual information was proposed. The proposed technique was shown to outperform previous approaches that did not exploit high-level audio events. Future extensions of this work include experimentation with additional measures for evaluating similarity in the audio event space, and the use of additional audio events, as well as other high-level audio-visual information, for further improving the accuracy of the results.

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, "On the use of audio events for improving video scene segmentation", Analysis, Retrieval and Delivery of Multimedia Content, N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Springer Lecture Notes in Electrical Engineering vol. 158, ISBN 978-1-4614-3830-4, 2012. The original publication is available at www.springerlink.com.

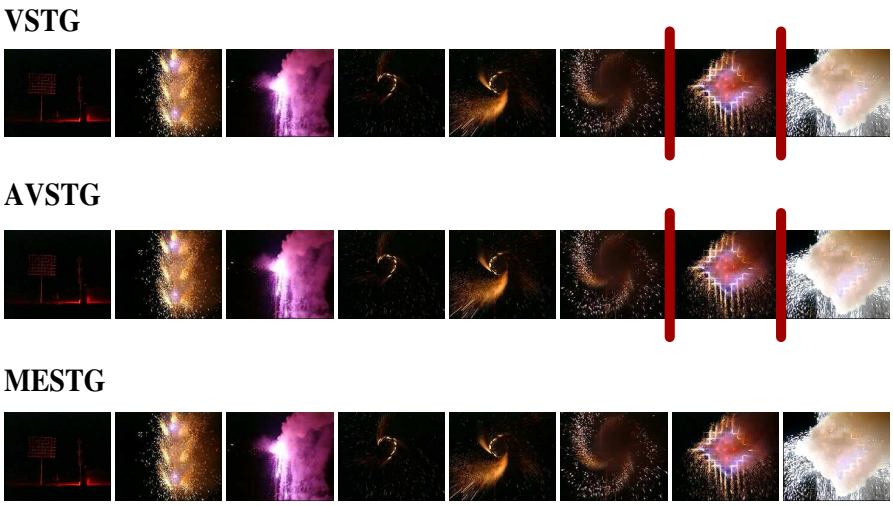


**Fig. 5.** A scene segmentation example. In each row, video shots are represented by one keyframe. These correspond to part of a single scene, formed by shots from a firework contest. No speech is contained in this part of the video; the audio content is limited to the sounds caused by the fireworks. As can be seen, both VSTG and AVSTG fail to recognize that the 7-th and 8-th shot also belong to the same scene with the rest of the shots, due to the fact that these are neither very similar in terms of appearance nor can be linked to the same speakers, in the absence of speech. On the contrary, MESTG manages to cluster all shots into a single scene, again demonstrating the significance of non-speech-related audio events.

## Acknowledgements

This work was supported by the European Commission under contracts FP6-045547 VIDI-Video and FP7-248984 GLOCAL.

## References

1. Tsamoura, E., Mezaris, V., Kompatsiaris, I.: Gradual transition detection using color coherence and other criteria in a video shot meta-segmentation framework. In: Proc. IEEE Int. Conf. on Image Processing, Workshop on Multimedia Information Retrieval (ICIP-MIR 2008). (2008) 45–48
2. Hanjalic, A., Lagendijk, R.L., Biemond, J.: Automated high-level movie segmentation for advanced video-retrieval systems. IEEE Trans. On Circuits and Systems for Video Technology **9**(4) (June 1999) 580–588
3. Yeung, M., Yeo, B.L., Liu, B.: Segmentation of video by clustering and graph analysis. Computer Vision and Image Understanding **71**(1) (July 1998) 94–109
4. Chasanis, V., Likas, A., Galatsanos, N.: Scene detection in videos using shot clustering and sequence alignment. IEEE Trans. on Multimedia **11**(1) (January 2009) 89–100
5. Nitanda, N., Haseyama, M., Kitajima, H.: Audio signal segmentation and classification for scene-cut detection. In: Proc. IEEE Int. Symp. on Circuits and Systems. Volume 4. (2005) 4030–4033
6. Chianese, A., Moscato, V., Penta, A., Picariello, A.: Scene detection using visual and audio attention. In: Proc. Ambi-Sys Workshop on Ambient Media Delivery and Interactive Television. (2008)
7. Wilson, K., Divakaran, A.: Discriminative genre-independent audio-visual scene change detection. In: Proc. SPIE Conf. on Multimedia Content Access: Algorithms and Systems III. Volume 7255. (2009)
8. Wang, J., Duan, L., Liu, Q., Lu, H., Jin, J.: A multimodal scheme for program segmentation and representation in broadcast video streams. IEEE Trans. on Multimedia **10**(3) (April 2008) 393–408
9. Sidiropoulos, P., Mezaris, V., Kompatsiaris, I., Meinedo, H., Trancoso, I.: Multi-modal scene segmentation using scene transition graphs. In: Proc. ACM Multimedia. (2009) 665–668
10. Amaral, R., Meinedo, H., Caseiro, D., Trancoso, I., Neto, J.: A prototype system for selective dissemination of broadcast news in European Portuguese. EURASIP Journal on Advances in Signal Processing **2007** (May 2007)
11. Meinedo, H.: Audio pre-processing and speech recognition for Broadcast News, PhD Thesis. IST, Technical University of Lisbon (March 2008)

12. Trancoso, I., Pellegrini, T., Portelo, J., Meinedo, H., Bugalho, M., Abad, A., Neto, J.: Audio contributions to semantic video search. In: Proc. IEEE Int. Conf. on Multimedia and Expo. (2009) 630–633
13. Bugalho, M., Portelo, J., Trancoso, I., Pellegrini, T., Abad, A.: Detecting audio events for semantic video search. In: Proc. Interspeech 2009. (2009)
14. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. (2001) Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.
15. Meinedo, H., Trancoso, I.: Age and gender classification using fusion of acoustic and prosodic features. In: Proc. Interspeech 2010. (2010)
16. Vendrig, J., Worring, M.: Systematic evaluation of logical story unit segmentation. IEEE Trans. on Multimedia **4**(4) (December 2002) 492–499