Improving Cluster Selection and Event Modeling in Unsupervised Mining for Automatic Audiovisual Video Structuring

Anh-Phuong Ta, Mathieu Ben, Guillaume Gravier

Improving Cluster Selection and Event Modeling in Unsupervised Mining for Automatic Audiovisual Video Structuring

Anh-Phuong TA[1], Mathieu Ben[2], and Guillaume Gravier[3]

[1] INRIA-Rennes, Campus Beaulieu, F-35042 Rennes, Cedex, France. Email: anh-phuong.ta@inria.fr
[2] Powedia, 12A avenue des Peupliers, F-35510, Rennes, Cedex, France. Email: mathieu.ben@powedia.com
[3] CNRS-IRISA, Campus Beaulieu, F-35042 Rennes, Cedex, France. Email: guillaume.gravier@irisa.fr

Abstract. Can we discover audio-visually consistent events from videos in a totally unsupervised manner? And, how to mine videos with different genres? In this paper we present our new results in automatically discovering audio-visual events. A new measure is proposed to select audio-visually consistent elements from the two dendrograms respectively representing hierarchical clustering results for the audio and visual modalities. Each selected element corresponds to a candidate event. In order to construct a model for each event, each candidate event is represented as a group of clusters, and a voting mechanism is applied to select training examples for discriminative classifiers. Finally, the trained model is tested on the entire video to select video segments that belong to the event discovered. Experimental results on different and challenging genres of videos, show the effectiveness of our approach.

Keywords: Video mining, Video structuring, Multimodality, Mutual Information, Event discovery, Structural event, Audiovisual consistency.

1 Introduction

The purpose of video structuring is to automatically find structural events in video sequences. Obviously, a structural element is represented as a key content of videos. Examples of such structural events are jingles in news videos, anchorpersons or participants in TV program videos, choruses from karaoke music videos, etc. Due to its potential applications in various fields, such as video summarization, video indexing and browsing, and content based video retrieval, video structuring can be considered as a crucial step in content-based video analysis. Existing methods can be broadly classified into two groups: (i) dense segmentation of the entire video, in which the video is mapped to a predefined structure [5] [6], and (ii) detection of a specific event like goals in sport videos [7] [8], advertisements, or anchorpersons [3] [4]. Despite their good results, most of these methods have two shortcomings: (a) they require manually annotated data for training models, thus these methods lack the generality to cope with diverse video sources; (b) they use clustering techniques to group similar video

segments, which form structural events. However, the problems that many clustering algorithms encounter are the choice of the optimal number of clusters, and how to deal with outliers.

It should be noted that there are very few research results available on discovering audio-visual events in an unsupervised way. Previous methods that are closely related to our work are those concerning video mining (video structuring). Video mining approaches [9] [10] [11] [12], which aim to detect regularly repeating patterns, have focused on the discovery of near-duplicate repetitions. However, these methods cannot deal with the structural events exhibiting content and temporal variations, i.e., the repetitions are not exact. There have been several works in unsupervised mining from videos [2] [5] [6]. The main idea of these methods is to exploit structural elements through mapping the entire video to predefined models. Because they are based on the assumption of dense segmentation, these approaches cannot handle the discovery of sporadic structural events.

In order to overcome the limitations mentioned above, in our previous work [1] we have proposed an unsupervised method to detect structural events from audio-visual documents without prior knowledge of the genre of the video. As presented in [1], two hierarchical clustering trees (called dendrogram) are first constructed for both audio segments and video shots (keyframes). We then measure the consistency between all pairs of audio-visual clusters by using mutual information (MI). Finally, several heuristics are used to select the best (i.e., the most relevant) audio-visual pair that represents a structural event. In this paper, we propose an extension to this method that makes it more robust to deal with variability and easier to extract multiple structural events. Compared to our previous work [1], the main contributions of this paper are summarized as follows:

- We present a new measure for selecting audio-visually consistent elements. The proposed measure is slightly different from the one used in [1], in that instead of using the original mutual information measure, which has 4 possible states (correlations) for two input binary variables, we use only two positive correlations (see section 3.1 for more details).
- We take into account the structure of the two given dendrograms to represent events, ie., each event is represented by a group of audio-visual cluster pairs, as opposed to only one pair used in [1], which may result in partial events (i.e. events whose occurrences are incompletely detected).
- In order to construct a model for each event, we propose a voting mechanism to select positive and negative examples as input vectors for Support Vector Machines (SVM). Note that, in [1], positive and negative examples are selected entirely based on the audiovisual segments of the best pair of clusters. Due to errors in clustering, however, not all these segments belong to the same structural event. In this work, we propose a more accurate method, in which each audiovisual segment will first cast its votes for negative or positive. Then, thresholds are used to select positive and negative examples (see section 3.3 for more details).

The rest of this paper is organized as follows. After briefly summing up the early work [1] for discovering structural events in section 2, we introduce our extensions of

this method in section 3. Section 4 describes experimental results. Finally, we conclude and give some perspectives of this work.

2 Discovering Audio-visually Consistent Events

In this section, we briefly summarize our early work on unsupervised video structuring [1], on which our approach is based. Figure 1 illustrates the main components of our approach, as well as the original approach[4] in [1]. Recall that our main objective was to design a generic approach to find events of interest that share common characteristics, including audio and visual presentations. First, the audio and video streams are respectively segmented into audio and video (shot) segments. We then extract commonly used audio and visual features (Gaussian components for audio, and RGB histograms for video) for segmentation and clustering. Classical bottom-up clustering algorithms are then applied for each modality to provide two different dendrograms representing video and audio clustering results. Given these two dendrograms, the work in [1] consists of 3 steps:

1. The first step measures mutual information (MI) between all pairs of audio and video clusters. These pairs of clusters are then ranked according to their MI values, and a list of n-best top pairs of clusters (i.e., the ones having highest MI values), called N-best list, is extracted.
2. The second one selects the best consistency pair among different pairs in the N-best list using several heuristics. This selected pair is considered to be representative of the most relevant event.
3. In the last step, based on the found event (the pair of clusters selected from the above step), positive and negative samples are extracted (see section 3.3 for more details). Then, a binary SVM classifier is applied to construct the event model.

In this paper, we aim at improving the cluster selection (step 1), and the event modeling techniques (step 3). We do not focus on using heuristics (step 2), which depend on the mining objectives, but we propose a new way to represent candidate events. The details of our approach is described in the next section.

3 Improving Cluster Selection and Event Modeling Techniques

Despite the promising initial results, the method introduced in [1] has several limitations:

a) It only allows to discover a single audio-visually consistent event.
b) It relies entirely on the consistent power of a single pair of clusters, i.e., a structural event is represented by a pair of audio-visual clusters only. Thus this method may result in partial structural events.

[4] This scheme, which illustrates the different components of our work, is slightly different from the original one presented in [1], but both operate on the same principle.

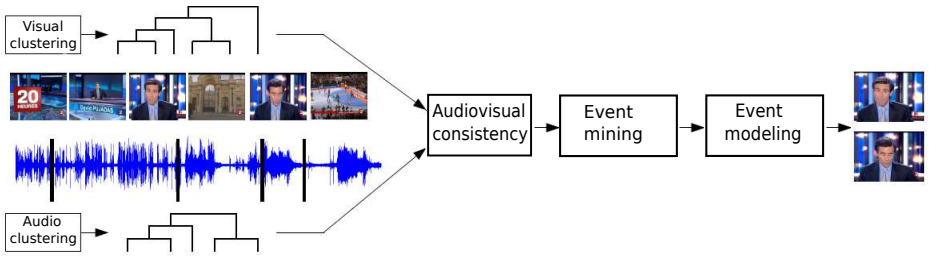

Fig. 1: Our general scheme for mining structural events from videos (this figure is adapted from [1]). In this figure, we show a simple example illustrating the detection of an anchor-person.

c) The original mutual information measure used in [1] to compute the consistency between two binary variables, which represent a pair of audio-visual clusters, is symmetric. That is, $MI(A,V) = MI(\bar{A},V) = MI(A,\bar{V}) = MI(\bar{A},\bar{V})$, where A, V are two binary variables denote the existence of audio, and video clusters, respectively; and $MI(\cdot,\cdot)$ is a function that measures mutual information of two given variables. This measure can give good results for detecting events that appear quite regularly, or for events, for which the corresponding audio and video segments from the selected clusters are quite strongly consistent. However, it cannot distinguish between relevant and irrelevant events (an irrelevant event appears when one of the two binary variables, or both is absent, i.e., for the cases of $MI(\bar{A},V)$, $MI(A,\bar{V})$, and $MI(\bar{A},\bar{V})$).

In this work we overcome these limitations by exploiting the N-best list based on the structure of the trees (i.e., dendrograms).

3.1 Estimating The Consistency of Audio-visual Clusters

After the hierarchical clustering of audio and video segments extracted from an input video, we obtain two corresponding dendrograms, where each node has a set of segments from the corresponding modality. Now we measure audio-visual consistency between an audio cluster and a visual cluster by using mutual information (MI) for two random variables. A large MI value indicates that the two corresponding clusters are closely consistent with each other and share more mutual information. In our case, for finding a repeating event, we aim at finding the consistency of audio-visual segments from a pair of clusters, which can be represented by the two positive correlations of the original mutual information. Let (C_i^A, C_j^V) be the i-th and the j-th nodes of the audio and video dendrograms, respectively. The mutual information between C_i^A and C_j^V is given as follows:

$$MI(C_i^A, C_j^V) = \sum_{(a,v)\in\{(0,0),(1,1)\}} p(a,v)\ln\left(\frac{p(a,v)}{p(a)p(v)}\right) \quad (1)$$

where a and v are binary random variables which respectively denote membership in C_i^A and C_j^V. The probabilities $p(a, v)$, $p(a)$, and $p(v)$ are estimated from the temporal distribution of segments. For instance, the join probability $p(a = 1, v = 1)$ is measured as the sum of the amount of time each segment of C_i^A co-occurs with a segments of C_j^V, normalized by the total duration of the video.

Equation 1 is applied for measuring MI between all audio and visual cluster pairs. Then, a list of pairs of clusters is established, and sorted in descending order according to its MI value. The objective of sorting is to help discover candidate events from more consistent to less consistent, and to filter out the irrelevant pairs (inconsistent pairs), i.e., the ones having low MI values. After removing inconsistent pairs, the final list remains n best pairs, called N-best list. In the next sub-section, we will analyze this list to discover structural events.

3.2 Event Mining

From the N-best list obtained in the above step, we extract candidate events. Due to the nature properties of hierarchical clustering algorithms, there are redundancies (in terms of structural relationships and similar characteristics) in the N-best list. Obviously, the pairs of clusters in the N-best list, which share redundant contents, should be part of an event (i.e., they should belong to the same structural event). More precisely, using a pair of audio-visual clusters (a, v) as example, if (a, v) represents a structural event, all other pairs of clusters from the two sub-trees constructed from (a, v) will represent (sub)instances (or other instances) of this event. Therefore, this event should be represented by a group of cluster pairs, including (a, v) and its *neighboring* pairs. See figure 2 for an explanation of this principle. Based on the analysis above, we extract multiple structural events in the following way: starting from the first pair in the N-best list (i.e., the one having highest MI value), we search from the N-best list to find all successor pairs from the two sub-trees constructed from this pair. The found successor pairs are grouped together with the current pair to establish a representative group for this event. This process is applied for each of the remaining pairs in the N-best list. The result is a list of group containing consistent audio-visual clusters, in which each group represents a structural event. The overall algorithm is presented in Figure 3, where $getSubtree(\cdot, \cdot)$ is a function that returns a subtree from a given node (cluster); $remove(\cdot, \cdot)$ function that removes a pair of clusters from a list; and $getFirstAV(\cdot)$ function returns the first element (ie., the pair of audio-visual clusters having highest MI value) from a list. Please note that we define a subtree for a given node as a list of nodes created from all paths that pass through it from a leaf to the root of the tree. An illustration example is shown in figure 2a, where the subtree for the red node is a list of nodes including all green nodes and the red node.

In our context, a consistent audio-visual cluster belongs to only one structural event. In other words, there are no structural events that share the same cluster pairs. Thus, after having constructed the list of groups, we remove for each group from this list, all common elements of any two groups. Consequently, the obtained list contains only non-intersecting groups, each of which represents a structural event. Note that although the algorithm presented in figure 3 returns all possible structural events from an input video, quantitative evaluations for multiple structural events require the knowledge of a

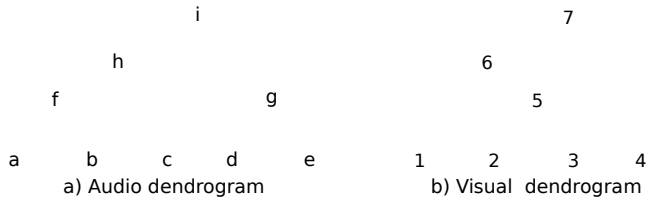

Fig. 2: Partial views of the audio (a) and visual (b) dendrograms, which illustrate the principle of the construction of a representative group for a given candidate event represented by a pair of clusters (see section 3): assuming that the pair of audio and video clusters in the N-best list (h,6) represents a candidate event. Then, the list of audiovisual candidate pairs contains: {(a,1) (a,2), (a,3),(a,5), (a,6), (a,7), (b,1), (b,2), (b,3), (b,5), (b,6), (b,7), (c,1), (c,2), (c,3), (c,5), (c,6), (c,7), (f,1), (f,2), (f,3), (f,5), (f,6), (f,7), (h,1), (h,2), (h,3), (h,5), (h,6), (h,7), (i,1), (i,2), (i,3), (i,5), (i,6), (i,7)}. From this list, a representative group for the candidate event is constructed by grouping together pairs that are present in the N-best list. Note that the yellow clusters from both dendrograms are not considered to be part of the representative group, because they may belong to another event.

Fig. 3: Algorithm for extracting groups of consistent audio-visual clusters.

Data: A ranked list of n pairs of clusters: nbestList; Two dendrograms corresponding to the two hierarchical clustering results: A, V;
Result: A list of groups containing consistent audio-visual clusters: E
k = 0;
while $not\ \ empty(nbestList)$ **do**
  $E(k) = \{\}$;
  $a, v \leftarrow getFirstAV(nbestList)$;
  $sub_a = getSubtree(A, a)$;
  $sub_v = getSubtree(V, v)$;
  for $\forall(a, b)\ |(a \in sub_a,\ b \in sub_b)$ **do**
    if $(a, b) \in nbestList$ **then**
      $E(k) = E(k) \cup (a, b)$;
      remove(nbestList,(a,b));
    end
  end
  $k = k + 1$;
end

human expert, and therefore out of the scope of this paper. In the next sub-section, we present a new method to model an event from a given group of audio-visual clusters.

3.3 Event Modeling and Recognition

After the event mining step, a candidate event is characterized by a group of audio and video clusters $E = \{e_1, e_2, ..., e_m\}$, where e_i represents a pair of audio and video

Fig. 4: Feature selection for SVM.

Data: A group of clusters E; thresholds T_p,T_n; set of audiovisual segments SEG
Result: Set of input vectors (including positive samples and the negative ones) for SVM: v_p, v_n
forall $s \in SEG$ **do**
  SN(s) $\leftarrow 0$;
  SP(s) $\leftarrow 0$;
end
foreach $e \in E$ **do**
  negVote(e, SN);
  posVote(e, SP);
end
$v_p \leftarrow (\boldsymbol{AV_{s_i}} \mid s_i \in SEG \; and \; SP(s_i) > T_p)$;
$doubtSet \leftarrow (s_i \in SEG \mid (0 < SP(s_i) \leq T_p \; and \; SN(s_i) > 0))$;
$v_n \leftarrow (\boldsymbol{AV_{s_i}} \mid s_i \in SEG \; and \; (SN(s_i) > T_n \; and \; s_i \notin doubtSet))$;

clusters (C^A, C^V) with the corresponding temporal segments (S^A, S^V). Note that the audio/video segmentation and clustering steps are performed independently for each modality. Thus S^A and S^V may be different from each other, and we need to combine them for building representative segments (i.e., audiovisual segments) of the pair in consideration e_i. For this end, an audiovisual segment of the video is constructed by merging the boundaries of an audio and a visual segments, and the corresponding feature vector (audiovisual feature vector) is the concatenation of the two component feature vectors. Recall that our goal here is to build a model based on audiovisual feature vectors, which can be used to predict a structural event from the entire video. Our early experiments [1] show that taking audiovisual segments belonging to the intersections of audio and video segments as positive examples[5], and those corresponding to neither their intersections nor their unions as negative examples, gives good results. Particularly, for the above pair e_i characterized by the two clusters (C^A, C^V), two sets of positive and negative training samples are determined as follows:

$$\boldsymbol{AV_{s_k}} \in +\mathbb{1} \text{ if } s_k \subset S^A \cap S^V$$
$$\boldsymbol{AV_{s_k}} \in -\mathbb{1} \text{ if } s_k \not\subset S^A \cup S^V$$

where s_k is an audiovisual segment, and $\boldsymbol{AV_{s_k}}$ is its corresponding audiovisual feature vector.

Now we extend this technique for the selection of training samples for the group of clusters E. The idea is as follows: for each element e_i (a pair of clusters) in E, we cast its votes for negative and positive samples (i.e., each audiovisual segment from e_i casts vote for positive or negative), and the voting results are accumulated for all elements in the group. This voting algorithm is described in figure 4, where $negVote(\cdot,\cdot)$ is a function that casts votes for negative examples for the audiovisual segments from a given pair of clusters. Similarly, $posVote(\cdot,\cdot)$ casts votes for positive examples. Note that,

[5] Note that, in our case we use a binary classifier to distinguish between an candidate event, i.e., positive class (+1), and non-candidate event, i.e., negative class (-1).

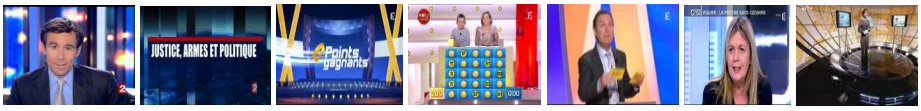

Fig. 5: Illustration of typical structural events in our dataset. From left to right: anchor person in news; a separator screen in flash news; a separator, two participants, and a presenter in games; a guest in a talk show; and magazine anchor person.

SN and SP are used to keep voting results accumulated over all elements in the input group. T_p, T_n are thresholds, which will be set experimentally, used to filter out the outliers. And $doubtSet$ is a set of audiovisual segments, where we do not have enough information to choose between negative and positive samples. Thus these segments should not belong to both negative and positive sets. Once the positive and negative samples are selected, discriminative classifiers can be used to train a binary model. In our work, we use a binary SVM with 5-fold cross-validation procedure on the training set to optimize Hyper-parameters. We then apply the trained model to all audiovisual segments in the input video. These segments are classified as corresponding to either the event under consideration or not.

4 Experimental Results

Discovering structural events based on audiovisual consistency should be evaluated on datasets, in which such events are present. Unfortunately, up to our knowledge no standard database is available for such difficult tasks. To evaluate our method, we captured TV programs with different genres from various French television channels (see fig. 5 for several examples of structural events), including: flash news, news, magazines, investigation reports, talk shows, and games. These videos are then annotated by a human expert. There are 18 video sequences (over 16 hours), in which the longest annotated event has about 120 occurrences and the shortest one has 9 occurrences. We used the classical recall, precision, and F-measure to evaluate the performance of our method. Note that, given the occurrences of an discovered event and its corresponding annotated event, recall is defined as the amount of time of correctly detected occurrences with respect to the total amount of time of all the occurrences from the corresponding annotated event in the ground truth, whereas precision is the amount of time of correctly detected occurrences with respect to the total amount of time of the detected occurrences. We performed two different experiments: the first was designed to evaluate the performance of the proposed measure for cluster selection. The second experiment aimed to evaluate the effectiveness of the proposed event modeling method. These two experiments are described below.

Evaluation of The Proposed Measure. In this evaluation, we compare the performance of the proposed measure for cluster selection with the baseline measure presented in [1]. For more convenience, we denote our measure by MI2 (i.e., it has two

possible correlations, cf. eq. 1), and the method in [1] by MI4. The testing protocol is as follows: for each video, both MI2 and MI4 are first applied to establish two corresponding N-best lists (ranked lists). Then SVM models are trained on samples extracted from the first element for each list. Note that ranking results in the N-best list provided by these two measures are not always in the same order. Therefore, for a fair comparison, for each genre of programs, we selected to report only results from the videos, for which both methods give exactly the same structural event corresponding to the first element in the N-best list. In the case that these two methods return the same event with different orders, we will discuss qualitative results later. Table 1 presents a comparison

Genre	MI4			MI2			#videos
	R	P	F1	R	P	F1	
Flash News	0.78	0.88	**0.83**	0.78	0.88	**0.83**	4/4
News	0.81	0.85	0.82	0.83	0.85	**0.84**	2/2
Magazine	0.91	0.97	**0.93**	0.91	0.97	**0.93**	2/5
Investigation	0.24	0.75	0.35	0.33	0.69	**0.44**	3/4
Games	0.71	0.71	0.71	0.79	0.72	**0.75**	2/3

Table 1: Comparison of the performance between MI4 and MI2.

of these methods, where the last column indicates the number of videos for which the two methods detect the same event (in terms of the first element in the two N-best lists) with respect to the total number of videos tested. For flash news, these two methods give exactly the same pair of audio and visual clusters, yielding the same results from SVM classification. This is due to the fact that such kinds of videos contain only one structural event and have little variations. From this table, we can observe that for the more challenging videos (eg., investigation videos), MI2 gives much better results. It should be noted that, except for flash news and magazines, MI2 always returns a pair comprising either the same audio cluster as MI4 and associated with a higher (i.e., higher level in the dendrograms) video cluster, or vice-versa. This indicates that MI2 is more robust to variability. Please note that, we cannot directly compare our results with those from the previous method [1], because the experimental set-ups are different, and we used more challenging videos for the tests in this work.

Evaluation of Event Mining and Modeling Techniques. In this experiment, we test the performance of the proposed event modeling method, i.e., evaluating the voting mechanism based on a representative group for a candidate event. To this end, we compare the performance for both MI2 and MI4 with and without using group (i.e., with or without using the proposed event mining and modeling techniques). For the case of without using group, the experimental set-up is the same as in experiment 1 above. For the case of using group, the testing protocol is given as follows: given the N-best list (obtained by MI2 or MI4), we apply the algorithm presented in Fig. 3 to obtain a list

Genre	Single pair			Group		
	R	P	F1	R	P	F1
Flash News	0.78	0.88	0.83	0.88	0.82	**0.85**
News	0.81	0.85	0.82	0.82	0.84	**0.83**
Magazines	0.78	0.87	0.82	0.79	0.87	**0.83**
Investigation	0.35	0.75	0.44	0.52	0.76	**0.57**
Talk shows	0.50	0.93	0.65	0.68	0.90	**0.72**
Games	0.63	0.81	0.68	0.75	0.76	**0.75**

Table 2: Comparison of the performance for MI4 with and without applying the proposed event modeling technique.

of representative groups of candidate events. The first group is then extracted from this list (i.e.., the group corresponding to the first element in the N-best list). For each group of audiovisual cluster pairs, we keep only 5 elements that have highest MI values. In all our experiments, thresholds[6] T_p is set to 1, and T_n is 4 for MI4, and T_p is set to 2, and T_n is 4 for MI2 (i.e., T_p and T_n are respectively set to be of 40% and 80% with respect to the group size). Finally, the algorithm presented in Fig. 4 is used to select training samples and SVM is applied to train the event model.

Table 2 and table 3 give the average results (in terms of recall, precision, and F1-measure) by using the measure introduced in [1] (MI4) and our measure (MI2), respectively. Where, the column, namely "Single pair", shows the obtained results from the best pair of clusters; and the column "group" shows the obtained results applying the proposed event modeling technique (i.e., the voting mechanism). From these tables, it can be easily seen that applying the proposed event modeling method (using a group of pairs) to represent events significantly outperforms the case of using only one pair of clusters (the most consistent pair in the N-best list). Taking the investigation videos for example, the performance of using the event modeling technique is increased by more than 10% with respect to that of using only one single pair. The results in terms of the average of F1-measure for MI2 using the voting method moderately decrease by roughly 1% for talkshows and news, however the corresponding recalls are higher. To evaluate the stability of our voting method, we also performed the tests for MI2 by varying the size of the group and setting T_p to 40%, and T_n to 80% of the group size, respectively, the performance changes by less than 1.5%.

Qualitative Analysis. Although quantitative analysis for multiple structural events is beyond the scope of this paper, we still performed the tests for this task, and observed qualitative trends of both MI4 and MI2 with and without applying the proposed event modeling technique. This allows us to point out some particular points: (a) using the proposed voting method gives much better results in general; (b) MI4 seems to be suited for sparse events, and events having little variations, and provides quite limited potential

[6] The choice of the optimal thresholds T_p and T_n is beyond the scope of this paper.

Genre	Single pair			Group		
	R	P	F1	R	P	F1
Flash News	0.78	0.88	0.83	0.84	0.88	**0.86**
News	0.83	0.85	**0.84**	0.85	0.83	0.83
Magazine	0.65	0.77	**0.71**	0.67	0.76	**0.71**
Investigation	0.45	0.65	0.49	0.61	0.66	**0.61**
Talk shows	0.65	0.85	**0.74**	0.67	0.80	0.73
Games	0.82	0.81	0.81	0.90	0.79	**0.83**

Table 3: Comparison of the performance for MI2 with and without applying the proposed event modeling technique.

for discovery of multiple events; (c) if both measures detect the same events but with different orders (the detected order from the N-best list), MI2 often returns a more complete event.

5 Conclusion and Discussion

In this paper, we have presented an improvement on cluster selection and event modeling in unsupervised mining for automatic audiovisual video structuring. The experimental results, using different genres of videos, demonstrate that the proposed method gives significant improvement compared to our previous work [1]. Our current work can be extended in several ways: first, we plan to evaluate multiple structural events, and to automatically determine the thresholds used for selecting training features. Second, we will explore how to automatically select n best pairs in the N-best list, and the relevant events among the different candidate groups. Finally, other features would be useful for the discovery of events in specific domains, eg., optical flows could be interesting for event discovery in sport videos.

Acknowledgments. This work was partly funded by OSEO, French State agency for innovation, in the framework of the Quaero research program. Many thanks to Monica Corlay for doing the ground truth annotation work, and to Sébastien Campion for providing Python code for key frame clustering, as part of the PimPy library[7].

References

1. Ben, Mathieu and Gravier, Guillaume. Unsupervised mining of audiovisually consistent segments in videos with application to structure analysis. IEEE International Conference on Multimedia and Exhibition ICME'11, Barcelona, Spain, July, 2011.

[7] http://pim.gforge.inria.fr/pimpy/

2. M. Naphade, C. Li and T. Huang. Discovering Recurrent Events in Multichannel Data Streams Using Unsupervised Methods. Chapter in the book "Data Mining: Next Generation Challenges and Future Directions". AAAI Press, 2004.
3. A. Hauptmann and R. V. Baron and M.Y. Chen and M. Christel and P. Duygulu and C. Huang and R. Jin and W. H. Lin and T. Ng and N. Moraveji and C. G. M. Snoek and G. Tzanetakis and J. Yang and R. Yan and H. D. Wactlar. Analyzing and searching broadcast news video. In Proc. of TRECVID,2003.
4. C., Tat-Seng and C., Shih-Fu and C., Lekha and H., Winston. Story boundary detection in large broadcast news video archives: techniques, experience and trends. In Proceedings of the 12th ACM international conference on Multimedia, 2004.
5. Clarkson, B. and Pentland, A. Unsupervised clustering of ambulatory audio and video. Proceedings of the Acoustics, Speech, and Signal Processing, on 1999 IEEE International Conference, Vol. 06, pp 3037-3040.
6. L., Xie and S., Chang and A., Divakaran and H., Sun. Unsupervised Mining of Statistical Temporal Structures. In Video mining, A. Rosenfeld et al. Eds, Chap. 10. Kluwer Academic Publishers, 2003.
7. Petkovic, M. and Mihajlovic, V. and Jonker, W. and Djordjevic-Kajan, S. Multi-Modal Extraction of Highlights from TV Formula 1 Programs. In: Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), 2002.
8. Fei Wang and Yu-Fei Ma and Hong-Jiang Zhang and Jin-Tao Li. A Generic Framework for Semantic Sports Video Analysis Using Dynamic Bayesian Networks. In International Multi-Media Modeling Conference, 2005, pp. 115-122.
9. M., Covell and S., Baluja and M., Fink. Detecting Ads in Video Streams Using Acoustic and Visual Cues. IEEE Computer Magazine, vol. 19, no. 12, 2006.
10. Cormac Herley. ARGOS: automatically extracting repeating objects from multimedia streams. In IEEE Transactions on Multimedia, vol. 8, no. 1, 2006.
11. Arne Jacobs. Using Self-similarity Matrices for Structure Mining on News video. In Advances in Artificial Intelligence, 4th Helenic Conference on AI, SETN 2006, vol. 3955 of LNAI, pp. 87-94.
12. Yang, Xian-Feng and Tian, Qi and Xue, Ping. Efficient Short Video Repeat Identification With Application to News Video Structure Analysis. IEEE Transactions on Multimedia, 2007, vol. 9, no. 3, pp. 600-609.