# Automatic Story Segmentation for TV News Video Using Multiple Modalities

Emilie Dumont, Georges Quénot

Hindawi Publishing Corporation
International Journal of Digital Multimedia Broadcasting
Volume 2012, Article ID 732514, 11 pages
doi:10.1155/2012/732514

*Research Article*

# Automatic Story Segmentation for TV News Video Using Multiple Modalities

**Émilie Dumont and Georges Quénot**

*UJF-Grenoble 1/UPMF-Grenoble 2/Grenoble INP, CNRS, LIG UMR 5217, 38041 Grenoble, France*

Correspondence should be addressed to Georges Quénot, georges.quenot@imag.fr

Received 18 November 2011; Revised 13 March 2012; Accepted 12 April 2012

Academic Editor: Werner Bailer



While video content is often stored in rather large files or broadcasted in continuous streams, users are often interested in retrieving only a particular passage on a topic of interest to them. It is, therefore, necessary to split video documents or streams into shorter segments corresponding to appropriate retrieval units. We propose here a method for the automatic segmentation of TV news videos into stories. A-multiple-descriptor based segmentation approach is proposed. The selected multimodal features are complementary and give good insights about story boundaries. Once extracted, these features are expanded with a local temporal context and combined by an early fusion process. The story boundaries are then predicted using machine learning techniques. We investigate the system by experiments conducted using TRECVID 2003 data and protocol of the story boundary detection task, and we show that the proposed approach outperforms the state-of-the-art methods while requiring a very small amount of manual annotation.

## 1. Introduction

Progress in storage and communication technologies has made huge amounts of video contents accessible to users. However, finding a video content corresponding to a particular user's need is not always easy for a variety of reasons, including poor or incomplete content indexing. Also, while video content is often stored in rather large files or broadcasted in continuous streams, users are often interested in retrieving only a particular passage on a topic of interest to them. It is therefore necessary to split video documents or streams into shorter segments corresponding to appropriate retrieval units, for instance, a particular scene in a movie or a particular news in a TV journal. These retrieval units can be defined hierarchically on order to potentially satisfy user needs at different levels of granularity. The retrieval units are not only relevant as search result units but also as units for content-based indexing and for further increasing the content-based video retrieval (CVBR) systems effectiveness.

A video can be analyzed at different levels of granularity. For the image track, the lower level is the individual frame that is generally used for extracting static visual features like color, texture, shape, or interest points. Videos can also be decomposed into shots; a shot is a basic video unit showing a sequence of frames captured by a single camera in a single continuous action in time and space. The shot, however, is not a good retrieval unit as it usually lasts only a few seconds. Higher-level techniques are therefore required to determine a more descriptive segment. We focus in this work on the automatic segmentation of TV journals into individual news or commercial sections if some are present. More specifically, we aim at detecting boundaries between news stories or between a news story and a commercial section. Though this work is conducted in a particular context, it is expected that it could be applied in some other ones with some adaptations, like talk shows for instance. Story segmentation allows better navigation within a video. It can also be used as the starting point for other applications such as video summarization or story search system.

We selected an approach based on multimodal feature extraction. The complementarities of visual and audio information from a video help to develop efficient systems. The story boundary detection is generally more efficient when several and varied features are used. The problem then

is to find the best way to use and combine such features. We use a temporal context and machine learning methods to perform the story boundaries detection from multiple features.

## 2. Related Works

Related works and existing solutions are developed in most cases for broadcast TV and more precisely for broadcast news. It was the case for the task proposed by TRECVID in 2003 and 2004 "Story segmentation" [1, 2] and the more recent ARGOS campaign [3]. Existing techniques for structuring a TV broadcast [4] are classified into three categories: manual approach by skilled workers, metadata-based approach, and content-based approach. We focus on the last category. The approach we explored is to segment at the story level; the video segmentation consists in automatically and accurately determining the boundaries (i.e., the start and the end) of each story.

The authors of [5] presented one of the first works on video segmentation in scenes. Their point of view for scene segmentation is to first locate each camera shot and second combine shots based on content to obtain the start and end points of each scene. They focus on low-level audio properties.

The method proposed by Chaisorn et al. [6, 7] obtained one of the best results at the TRECVID 2003 story boundary detection task as they achieved an $F1$ measure accuracy over 0.75. They first segmented the input video into shots. They then extracted a suitable set of features for modeling the shots contents. They employed a learning-based approach to classify the shots into the set of predefined categories. Finally, they identified story boundaries using a HMM model or inductive rules. However, they selected 13 categories of shots, like Anchor, Sports, Weather, Program logo, Finance, and Speech/Interview. Although effective, their technique required a lot of manually annotated data. The method proposed here requires much less annotated data.

Recently, the authors of [8] segmented videos into stories by detecting anchor person in shots; the text stream is also segmented into stories using a latent-Dirichlet-allocation-(LDA-) based approach. They obtained an $F1$ measure equal to 0.58 on the TRECVID 2003 story boundary detection task. In the paper [9], they presented a scheme for semantic story segmentation based on anchor person detection. The proposed model makes use of a split and merge mechanism to find story boundaries. The approach is based on visual features and text transcripts. The performance of this method is over 0.6 for $F1$ measure also on the TRECVID 2003 story boundary detection task. In the study [10], a set of key events are first detected from multimedia signal sources, including a large-scale concept ontology for images, text generated from automatic speech recognition systems, features extracted from audio track, and high-level video transcriptions. Then, a fusion scheme is investigated using the maximum figure-of-merit learning approach. They obtained an $F1$ measure equal to 0.651 on the TRECVID 2003 story boundary detection task.

In this paper, we propose a more effective method than the actual state of art (evaluated on the same test data). Moreover, our method requires a minimal annotation effort. Though it requires a development set including a number of representative videos with a story segmentation ground truth for training, it does not require or requires very little additional feature annotation like the presence of anchorpersons in shots or of topics like sports, weather, politics, or finance for instance.

## 3. System Overview

*3.1. News Structure.* Most news videos have rather similar and well-defined structures. Chaisorn et al. [7] have studied the structure of news videos and noticed:

> *"The news video typically begins with several Intro/Highlight video sequences that give a brief introduction of the upcoming news to be reported. The main body of news contains a series of stories organized in terms of different geographical interest, such as international, national, regional and local, and in broad categories of social political, business, sports and entertainment. Each news story normally begins with an anchorperson. Most broadcasts include reports on Sports, Finance or Weather."*

There are also, depending or the station, sequences of commercials. Figure 1 illustrates the structure of a typical news video. Although the ordering of news items may differ slightly from broadcast station to station, they all have similar structure and news categories.

*3.2. Choice of a Segmentation Unit.* Most of the previous works used the shot as a basic segmentation unit for performing story segmentation. However, we noticed that in the TRECVID development set, only 94.1% of the story boundaries match a shot boundary with the 5-second fuzziness allowance of the official evaluation metric. This means that a system working at the shot level cannot find about 6% of the story boundaries. For example, at the end of a story, an anchorperson can appear to give a summary or a conclusion and switch to another topic. In this case, there is no shot transition between the two stories.

On the other hand, the individual frame is a much too small unit not only because of the volume of computations involved by a frame-level evaluation but also because such an accuracy is not required at the application level and because we felt that the segmentation unit should be long enough so that it has a visual meaning when seen by a human being. It was demonstrated during the TRECVID task of Rushes Video Summarization in 2007 that one second is a good duration for a video segment to be meaningful. Two papers showed, in parallel, that one second is enough and sufficient to represent a topic [11, 12].

We finally decided to use a short duration and fixed-length segmentation for the story boundary candidate points and for the segment contents characterization. In preliminary experiments, we also tested segment durations larger


Figure 1: The structure of a typical news video.

than one second and the best results were obtained with the smaller ones. We consequently decided to use one second, as the basic unit, which is also consistent with previous works on video summarization [13]. One-second accuracy on story boundary location is also enough from an application point of view.

*3.3. Global System Architecture.* The idea of our approach is to extract a maximum of relevant information (features or descriptors) and then to fuse it for detecting transitions between the stories. Figure 2 shows the proposed scheme.

Relevant information is extracted on all one-second segments. We use a classification process on the basic units but only in an unsupervised way. Classifying the video segments into different classes (anchorperson, logo presence, weather, speech, silence...) is a fundamental step in recovering structure of a news program. Within a story, we assume that the environment is similar and the discussion focuses on the same topic.

We decided to use the different available modalities. The visual information includes shot detection, the presence of a particular person, and other information such as the presence of channel logo, junk frames, and visual activity. We also use the presence of screen text; we believe that the presence of a text box on a frame on a particular location may have some importance to find story boundaries. For example, in television news the title of a new topic appears in the same place.

We extract audio information like the presence of silence. In fact, when an anchorperson speaks, it happens regularly that a short silence marks the transition between two topics. We also exploit automatic speech recognition (ASR) to extract textual information such as the presence of words that appear frequently near a transition between the stories.

One originality of the proposed approach is that once extracted, the descriptors are expanded with a local temporal context. The main idea of this step is that the value of a descriptor is a possible cue for a story boundary but its temporal evolution in the neighborhood is possibly also very relevant. For example, the appearance or disappearance of a logo is an information more important than only the presence of the logo in the video sequence. Now that we have different sources of information, we need to merge them in order to predict the story boundaries. These sources are merged by early fusion [14].

Once we have different sources of information for each one second segment as well as their local temporal evolution, the challenging task is to segment the broadcast into coherent news stories. Like in major works, we focus on finding the boundaries of every story that succeed in the video stream. In order to perform this detection, we use traditional machine learning methods.

## 4. Multimodal Features-Based News Stories Segmentation

We present in this section the extraction of the different features. These features are either obtained directly through the application of a third party system that we could not have a chance to improve (e.g., the speech recognizer system (Section 4.2.2)) or built for our purpose (e.g., the anchor person detector (Section 4.1.2)). Application of a text tiling method [15] on the speech transcription was also considered but, surprisingly, it was found that it did not help.

### *4.1. Visual Features*

*4.1.1. Shot Detection.* We perform a shot boundary detection. As explained previously, in TRECVID development set, 94.1% of the story boundaries appear near a shot boundary. Therefore, this information is very important. Shot boundary detection has been performed by using the system described in [16]. This system detects cut transitions by direct image comparison after motion compensation and dissolve transitions by comparing the norms of the first and second temporal derivatives of the images. It also contains a module for detecting photographic flashes and filtering them out as erroneous cuts and a module for detecting additional cuts via a motion peak detector. This system obtained an overall of recall/precision of 0.819/0.851 at the TRECVID 2003 evaluation campaign. More precisely, it obtained a recall/precision of 0.91/0.92 for cut transitions and of 0.72/0.88 for gradual detection.

Shot boundary detection is performed but it is not directly used as a basis for the candidate story boundaries as this would induce a significant number of missed transitions (at least 6% of story boundaries do not match a shot boundary). Instead it is used as a feature associated to one-second segment units: two binary values are associated with each one-second segment indicating the presence or the absence of a cut or gradual transition within it.

*4.1.2. Face and Anchor Person Detector.* We use a face detector [17] for which the authors report a face detection rate of 0.903. In order to detect anchor person sequences, we assume that frames with the anchor person are frames that (i) contain a face centered and (ii) are very likely to appear frequently almost "as is" in the video. Consequently, we first select the frames that contain a centered face as candidates to be an anchor person template for a given video. The face being frontal and rather static, in this case the face detector is reliable. For a given video and in order to select an appropriate anchor person template, we expect the average visual similarity of candidates with a prefixed percentage of candidates to be maximal and choose the template frame

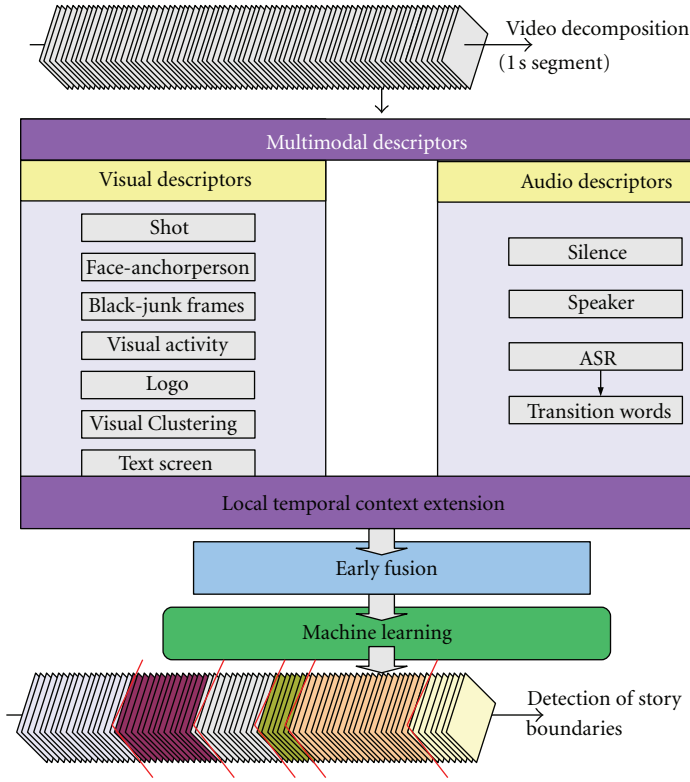


Figure 2: Overall system components.

as the frame that exhibits the greatest similarity; Figure 3 shows template samples. The similarity used is based on a Euclidean distance computed between color histogram in the HSV color space (18-3-3 bins). Finally, the similarity between the template and a frame is used like a confidence measure of the presence of an anchor person. Note that preliminary experiments without the face preselection can be used as a plateau detector. This detector has been evaluated on a collection of French TV news videos for anchor person detection on which we obtain an $F1$ measure equal to 0.865 with a recall equal to 0.857 and a precision equal to 0.891.

The anchorperson feature is a single analog (real) value associated with each one-second segment, which is the confidence measure for the segment to contain the anchor person.

*4.1.3. Junk Frames.* A junk frame is a noninformative frame, typically strong compressions artifacts, transmission errors, or more simply black or single color frames. Figure 4 shows examples of junk frames. A junk sequence may inform us about the possibility of finding a transition around this sequence. Despite its apparent simplicity, the problem of junk detection is quite delicate. We propose a two-step method: detecting black frames and detecting single-color frames. We detect single-color frames by computing the entropy of the distribution of color pixels in gray color space and we remove frames with entropy lower than a predefined threshold. The junk frame feature is a single analog value associated with each one-second segment indicating the likeliness of the segment to contain one or more junk frames.

*4.1.4. Visual Activity.* The intensity of motion activity in a video is in fact a measure of "how much" the video content is changing. Considering the high computational complexity caused by existing methods to model the motion feature, we use a more computationally effective color pixel difference-based method to extract the visual activity. The visual activity of a frame can be represented by the percentage of pixels that have changed color between it and the previous frame.

*4.1.5. Logo Detection.* A TV logo is a graphic representation that is used to identify a channel. A logo is placed in the same place and continuously, except during commercials. Based on this observation, we compute the average frame of the video and the variance of the pixel color in the video, see Figure 5. Pixels with the lowest variance are considered to be part of the logo. Their position will be called the reference position. During the logo detection step, for a given frame, the absolute difference between the colors of the pixels situated at a reference position and their counterpart in the average image is computed. The lower the sum is, the more probable the logo is in the frame. We manually selected the search region for each different channel only to reduce the computational time. However, this method works when applied on the whole image. A temporal filter is applied to the estimated probability of the presence of the logo. This filter outputs a binary value for each one-second segment indicating the presence or the absence of the logo within it.

Figure 3: Samples of anchorperson template.

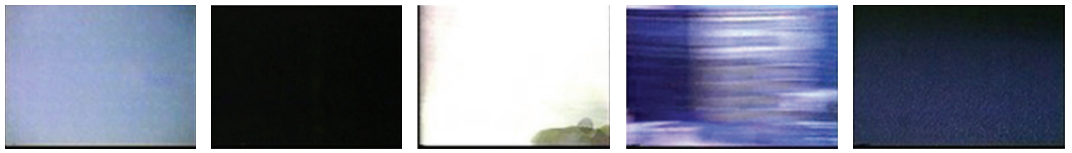

Figure 4: Samples of junk frames.

Figure 5: Average frames and reference position; CNN images on the right and ABC on the left. The first image represents the average frame (for a selected location), and, on the second image, the pixels with the lowest variance in white are considered to be part of the logo.

*4.1.6. Screen Text.* The screen text boxes are detected using the method proposed in [18]. Several successive filters are passed on frames: a Sobel filter to determine character edges, then treatment of dilatation and erosion connects the characters together. The connected components that do not hold a mandatory geometry are filtered. Detection is performed on each frame and only the boxes sufficiently stable over time are kept. We only use the presence of a text box information because the quality of many videos is too poor for a good optical character recognition. The text feature is a single analog value associated with each one-second segment indicating the likeliness of the segment to contain one or more text boxes.

*4.1.7. Visual Clustering.* We perform a clustering in order to group video segments by visual similarity. We represent a video segment by an HSV color histogram, we use the Euclidian distance to compare video segments, and finally we use K-means to perform the clustering. The cluster feature is a discrete integer value associated with each one-second segment indicating the index of the cluster that is closest to the segment contents.

*4.2. Audio Features*

*4.2.1. Silence Detection.* The first step of audio segmentation systems is to detect the portions of the input audio stream that exhibit some audio activity or, equivalently, the portions of silence. The approach for audio activity detection is the bi-Gaussian model of the stream energy profile, where the energy profile is the frame energy or log-energy sequence. The silence feature is a binary value associated with each one-second segment indicating that the segment does contain silence.

*4.2.2. Automatic Speech Recognition (ASR).* We used here the transcripts proposed during the TRECVID 2003 story segmentation campaign. The speech recognizer makes use of continuous density HMMs with Gaussian mixture for acoustic modeling and 4-gram statistics estimated on large text corpora. Word recognition is performed in multiple passes, where current hypotheses are used for cluster-based acoustic model adaptation prior to the next decoding pass [19]. In our context (i.e., broadcast news), the speech recognizer has a word error rate of 14%. ASR is not directly used for producing a feature. Text tiling was tried on it but it was not able to lead to an overall improvement. However, transitions words extracted from it have been found useful.

*4.2.3. Speaker Detection.* The speaker detection method is based on [20]. The system used the normalized cross likelihood ratio (NCLR). First, the NCLR is used as a dissimilarity measure between two Gaussian speaker models in the speaker change detection step, and its contribution to the performance of speaker change detection is compared with those of BIC and Hostelling's T2-Statistic measures. Then, the NCLR measure is modified to deal with multi-Gaussian adapted models in the cluster recombination step. This step ends the step-by-step speaker diarization process after the BIC-based hierarchical clustering and the Viterbi resegmentation steps. The speaker diarization error obtained by this method was 7.3%. The speaker feature is a discrete integer value associated with each one-second segment indicating the index of the speaker present in the segment.

*4.2.4. Transition Words.* Based on the ASR, we extract the most frequent transition words. We first remove all stop

Table 1: Transition words and their scores.

| Words | $t-3$ | $t-2$ | $t-1$ | $t$ | $t+1$ | $t+2$ | $t+3$ |
|---|---|---|---|---|---|---|---|
| ABC | 0.02 | 0.03 | 0.016 | 0.01 | 0.12 | 0.62 | 0.18 |
| News | 0.03 | 0.16 | 0.15 | 0.04 | 0.29 | 0.33 | 0.06 |
| Tonight | 0.07 | 0.23 | 0.32 | 0.10 | 0.14 | 0.10 | 0.04 |
| Today | 0.18 | 0.30 | 0.46 | 0.02 | 0.00 | 0.01 | 0.02 |

words from the transcription. Then, we select the most frequent words that appear in a temporal window that overlaps a story transition. Finally, for each selected word $w$, we determine a score related to the nonuniform probability to find a transition at time $t + i$ sec given that $w$ were pronounced at time $t$.

Table 1 shows results obtained on ABC videos. If $i$ ranges between $-3$ and $+3$ seconds, we can notice that the extracted words are ABC, News, Today and Tonight, ABC and News being pronounced one or two seconds after a transition while Today, and Tonight appear a few seconds before a transition. The transition word feature is a single analog value associated with each one-second segment giving the probability of the segment to correspond to a story boundary according to the presence of a possible transition word in its neighborhood.

*4.3. Multimodal Features.* Multimodal features are the pool of features obtained from single modalities to be used for story boundary detection combined into a global representation. Figure 6 shows a graphical representation of multimodal features. This figure is quite complicated but it is very useful to see the various shapes and the complementarities of the individual features. The multimodal features correspond to a concatenation of all the elements within one column (early fusion) before the local temporal extension.

As it can be seen, silence is well correlated with the ground truth although it lacks precision (it detects a silence between the first two story boundaries). This false alarm can nevertheless be corrected using other features like, for example, anchorperson or shot transition. The combinatorial is very complex, so we rely on an automatic procedure to combine these features and machine learning to analyze them.

The shot detection information is decomposed into two binary values: the first one represents the presence of a cut transition and the second represents the presence of a gradual transition in the one-second segment. The presence of silence and logo are represented by a binary value. Visual cluster and speaker are represented by the cluster index. Finally, other features are numerical values.

## 5. Multimodal Fusion

Once extracted, the multimodal features can be combined by early fusion in order to detect the transitions between stories. We do this in two steps: we determine the best way to use each feature and then we merge the features using a classifier. The classifier provides a prediction score for story transition. The fusion is performed with the same basic segmentation unit as the feature extraction: one-second fixed length segments.

*5.1. Local Temporal Context.* All descriptors are extracted for each one-second segment of a video. Therefore, they do not take into account the temporal information included in a video. Certainly, the information of the presence or absence of a descriptor is important, but the information about the appearance or disappearance can be even more relevant. Based on this observation, we extend the descriptors with a local temporal context, more precisely by the descriptor values in the closest segments.

We use a strategy based on a sliding window: for a one-second segment $s$ coming into sight at time $t$ in the video, we use a sliding window with a fixed length equal to $2l + 1$ and where the current segment is located at the center of the window $W_s = \{s_{t-l}, \ldots, s_t, \ldots, s_{t+l}\}$. For each sliding window, we extract three categories of representations:

(i) the list, $V_{\text{all}}$, of all values contained in the sliding window ($2l + 1$ values);

(ii) the list, $V_{\text{diff}}$, of the differences between each couple of one second segment with an equal distance to $s_t$ plus the central value $s_t$ itself ($l + 1$ values);

(iii) $V_{\text{gauss}}$, the values of the Gaussian distribution, the derivation of Gaussian distribution, and the second derivation of Gaussian distribution (3 values).

The first solution corresponds to feeding the classifier with an input vector that is a concatenation of a number of column vectors around the current one or to use a vertical slice of several columns in the representation given in Figure 6. This is the most complete information that can be passed on and it leaves open to the classifier underlying machine learning method to decide whether it will use for each feature the single central value, the level around it, the variation around it, or any combination of them including how far around it should go. Though this is the most complete, it is also the most costly one and not necessarily the most efficient one. As we can have the intuition that the level, the variation, or a combination of both can be more compact and more synthetic we considered the two other possibilities, the third one being even more compact and synthetic than the second one. We also considered the possibility of optimizing the size of the window and the neighborhood representation type by tuning them using a development set.

*5.2. Fusion.* Finally, each multimodal vector used as input for the classifier is a concatenation of the best features' representation. We chose to perform an early fusion for avoiding the loss of the correlation information between different features. We have tested several classifiers using WEKA [21] to find the best one for the task. In contrast with a shot transition, a story transition is not necessarily annotated at the same frame for all annotators. In order to take into account the fuzziness of the story transition location in the annotation, we decided to discard five one-second segments on each side of a story transition since

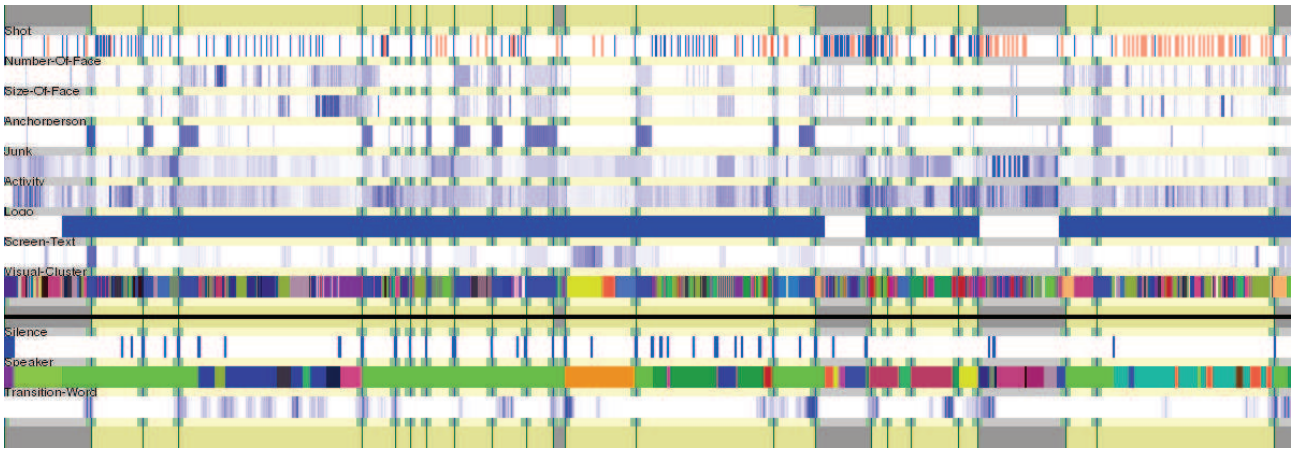


Figure 6: Example of multimodal features. Each pixel column corresponds to a one-second segment. The top and bottom thick lines (or stripes) represent the ground truth with transitions in black and stories in light green (news) or dark gray (advertisements/misc). The similar line (or stripe) with a thick black line in the middle shows the same information while also separating the visual features (above) from the audio features (below). Thin lines between the thick ones reproduce the top and bottom thick lines but with lighter colors for the story types and additionally with a 5-second green expansion around the boundaries corresponding to the fuzziness factor associated with the evaluation metric (transitions are counted as correct if found within this extension). These are replicated so that it is easier to see how the feature values or transitions match them. Also, the beginning of the thin lines contains the name of the feature represented in the thick lines immediately below them. Finally, the remaining thick lines represent the feature values with three types of coding. For scalar analog values, the blue intensity corresponds to the real value normalized between 0 and 1. For binary values, this is the same except that only the extreme values are used and that in the case of shot boundaries, blue is used for cuts and red is used for gradual transitions. For cluster index values (clusters and speakers), a random color map is generated and used.

these segments are annotated negatively while they could be positive and since such outliers often lead to a loss of performance. Discarding these segments ensures that all the samples annotated as negative are actually negative while those annotated as positive are chosen as close to the actual transition as possible. This might in turn result into a comparable fuzziness into the location of the detected transitions.

## 6. Experimental Results

*6.1. Experimental Protocol.* Our method has been evaluated in the context of the TRECVID 2003 Story Segmentation Task and exactly in the same conditions except, indeed, that it was done later and that it could not be included in the TRECVID 2003 official results. However, the same data, ground truth, protocol, metrics, and evaluation programs have been used. Tuning has been done using only the development data and the tuned system has then been applied only once on the test data. No tuning was done on the test data at all.

The collection contains about 120 hours of ABC World News Tonight and CNN Headline News recorded by the Linguistic Data Consortium from late January through June 1998. We chose this dataset because it is the only one which is available and widely used by the community; it allows us to compare our method with the state of the art.

We developed and tuned the system only within the development set (partitioned itself into a training and a test set by a random process) and then we applied it on the test set. Since story boundaries are rather abrupt changes of focus, story boundary evaluation is modeled on the evaluation of shot boundaries: to evaluate the story segmentation, an automatic comparison to human-annotated reference is done to extract recall and precision measures. A story boundary is expressed as a time offset with respect to the start of the video file in seconds, accurate to the nearest hundredth of a second. Each reference boundary is expanded with a fuzziness factor of five seconds in each direction, resulting in an evaluation interval of 10 seconds. If a computed boundary does not fall in the evaluation interval of a reference boundary, it is considered a false alarm.

(i) Story boundary recall = number of reference boundaries detected/total number of reference boundaries.

(ii) Story boundary precision = (total number of submitted boundaries minus the total amount of false alarms)/total number of submitted boundaries.

(iii) Story boundary $F1$ measure = 2 × recall × precision/(recall + precision).

*6.2. Classifier Selection.* We made a selection of the best classifier method for our problem: 48 classifiers from the Weka toolbox have been tested; for more information about these classifiers, see [22]. Figure 7 shows the best results obtained in terms of $F1$ measure within the development set.

Results show that RandomForest is the best classifier for our problem. Results also show that the classifiers in the category of trees are on average the best in our case. This can partially be explained by the non-normalized features that we used. However, this is a complex problem because our descriptors do not have the same scale. For example, it is

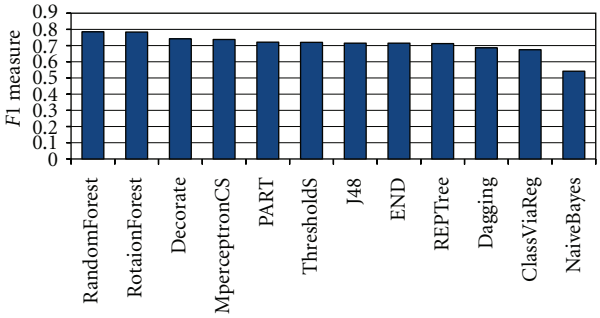


Figure 7: Results for the best classifiers.

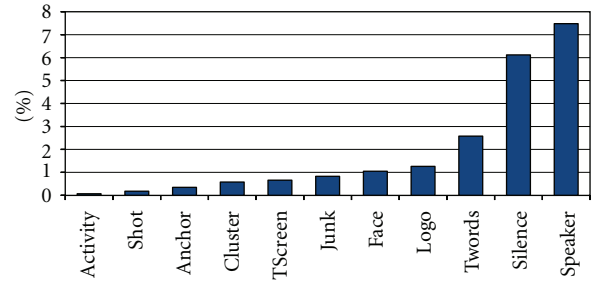


Figure 8: Multimodal features lost in terms of $F$ measure.

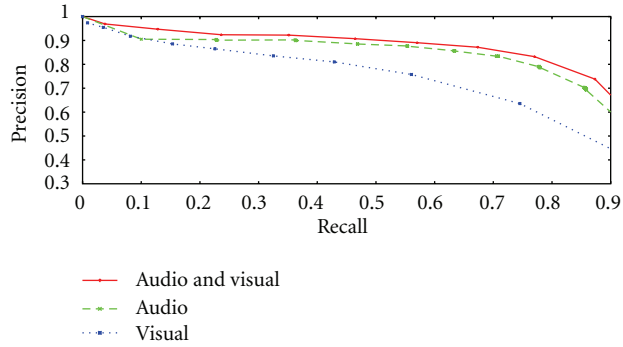


Figure 9: Comparison between audio and visual features.

difficult to compare the number of faces in a video segment and a confidence value of visual activity. For our problem, it is also interesting to note that the amount of positive is very low compared to the number of negative. So, classifiers like SVM are not suitable.

*6.3. Feature Interest.* To prove the relevance of the chosen features, we estimate the performance loss in terms of $F1$ measure when features are individually removed from the pool. We train several classifiers by removing one feature at each time. Finally, we compare their result with those obtained by the method taking into account all features, see Figure 8.

We can see that speaker detection and silence are the most important features for our problem. Features like transition words, logo, face, junk, text screen, and visual cluster are also important. It should be noted that some features are correlated with other ones, and it is logical that the performance loss associated with such a feature is not high. For example, if we remove anchorperson, the performance loss is not very important because this information is partly present in the speaker feature.

We can see that audio features are more interesting than visual features. In order to evaluate this comment, we compare results obtained only using audio features with only visual features, see the recall-precision curve in Figure 9. It is clear that audio features are better; therefore visual features improve results.

*6.4. Local Temporal Context Experiments.* For each descriptor, we tested different lengths of sliding window (from 1 sec to 31 sec) and different representations ($V_{\text{all}}$, $V_{\text{diff}}$, or $V_{\text{gauss}}$) in order to find the best combination for each descriptor (other descriptors were used without local context information). Figure 10 shows the results for three different descriptors: speaker and face. The curve Base represents results without local temporal extension. It is clear that the local temporal context improves the quality of the predictions. Table 2 shows the best combination for the selected descriptors.

In Figure 11, we compare the performance of the different methods of local temporal context extractions. We can see that the local temporal context improves performance, and the best results are obtained by using the best local temporal context for each descriptor. This method uses vectors of 231 dimensions. The closest results to this method are obtained using a sliding window equal to 15 and extracting $V_{\text{all}}$, $V_{\text{diff}}$, and $V_{\text{gauss}}$ for each descriptor; however, in this case, the dimensions become 650. So the selection of optimal parameter for each descriptor is more interesting.

*6.5. Cross-Channel Experiments.* In order to assess the robustness of our system, we evaluate it in a cross-channel setting while the domain being the same (namely, TV news programs). The TRECVID 2003 collection contains TV journals from two different channels: CNN and ABC. We evaluated the system while training the system on the full development collection, only the ABC part, or only the CNN part and while testing the system also on these channel combinations. In order to distinguish between the effect of using a smaller training collection and the effect of using only one of the channels, we also trained the system using only half of the full development collection with both channels. We evaluated the following combinations: "ABC to ABC," "CNN to CNN," "all to all," "all/2 to all," "ABC to CNN," and "CNN to ABC". Some features (logo detection and transition words) are always computed separately for each channel.

Figure 12 shows the results of the cross-collection examination. The system has a very stable behavior when the composition of the training and test data is similar. The performance is slightly higher for "ABC to ABC" and slightly lower for "CNN to CNN," possibly indicating that the organization of ABC journals is more stable than that of CNN journals. The size of the training set has no significant effect.

Table 2: Best descriptor representation. In this table, we can see for each descriptor the best length $l$ for the sliding window and the selected categories of values.

| | Shot | Anchor | Silence | Speaker | Face | TWord | TScreen | Junk | Activity | Logo |
|---|---|---|---|---|---|---|---|---|---|---|
| Length | 1 | 21 | 9 | 15 | 11 | 13 | 5 | 9 | 13 | 21 |
| Values | $V_{all}$ | $V_{diff}$ | $V_{diff}$ $V_{gauss}$ | $V_{diff}$ | $V_{all}$ | $V_{all}$ $V_{gauss}$ | $V_{all}$ | $V_{diff}$ | $V_{diff}$ | $V_{diff}$ $V_{gauss}$ |

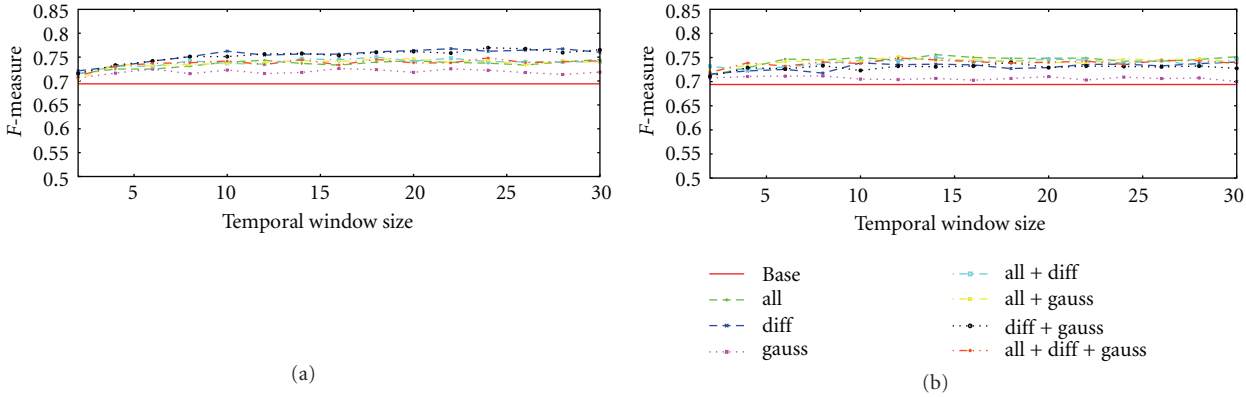


Figure 10: Results for local temporal context of a descriptor.

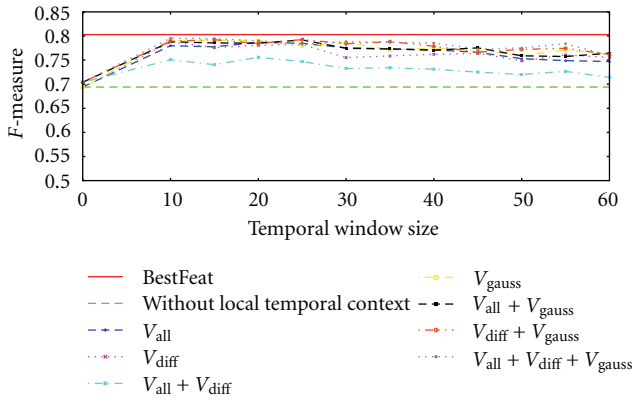


Figure 11: Results for local temporal context.

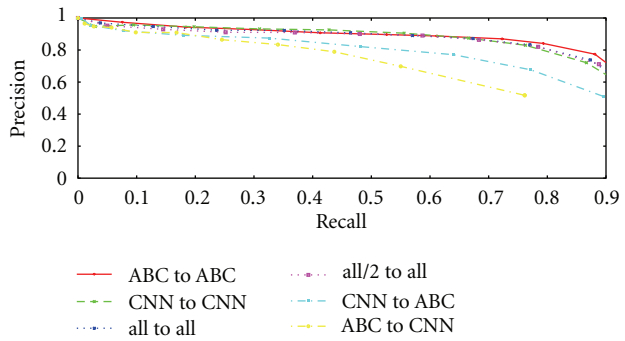


Figure 12: Collection results. The comparison of results between a learning on all videos (CNN and ABC) called "all to all," a collection learning "ABC to ABC" and "CNN to CNN," and another generic learning but with the same number of training samples "all/2 to all" as in collection learning.

As expected, we can notice a performance drop for cross-channel experiments. The figure shows that the system performs better for "CNN to ABC" than for "ABC to CNN." However, the quality of the predictions remains good since we get an $F$ measure of 0.696 (recall = 0.642, precision = 0.761). The difference in performance between CNN to ABC and ABC to CNN probably arises from the fact that CNN has the same style of transitions as ABC but CNN also contains specific transitions not observed in ABC.

*6.6. Experiments with Another Corpus.* We have tested our method on another corpus. This corpus consists of 59 videos of France 2 TV News from 1 February to 31 March 2007. The average length of these videos is about 38 minutes, which represents an overall of 37 hours of video. We extracted a subset of multimodal features: junk frames, visual activity, logo, anchorperson, transition words, and speaker detection. We obtained good results: an $F1$ measure equal to 0.870 with a recall equal to 0.897 and a precision equal to 0.844. Our method applies well to another corpus, no adjustment has been made, and the system has been applied as such with the descriptors we had. One reason for this good performance can also come in the quality of videos and probably the story boundaries are easier to predict than on TRECVID 2003 corpus.

*6.7. Results.* We compare our results with the state of the art in Table 3.

(i) The method proposed by Chaisorn et al. [6, 7] obtained one of the best results at the TRECVID 2003 story boundary detection task. They, first, segmented the input video into shots. Then, they extracted a suitable set of features to model the contents of shots.

Table 3: Comparison with the state of art.

| | Chaisorn et al. 2003 [7] | Misra et al. 2010 [8] | Goyal et al. 2009 [9] | Ma et al. 2009 [10] | Our method | Our method + channel |
|---|---|---|---|---|---|---|
| Recall | 0.749 | 0.54 | 0.497 | 0.581 | 0.878 | 0.893 |
| Precision | 0.802 | 0.64 | 0.750 | 0.739 | 0.767 | 0.767 |
| $F1$ | 0.775 | 0.58 | 0.600 | 0.651 | 0.819 | 0.825 |

They employed a learning-based approach to classify the shots into the set of 13 predefined categories. Finally, they identified story boundaries using an HMM model or inductive rules.

(ii) Misra et al. [8] segmented videos into stories by detecting anchor person in shots; the text stream is also segmented into stories using a latent-Dirichlet-allocation- (LDA-) based approach.

(iii) Goyal et al. [9] presented a scheme for semantic story segmentation based on anchor person detection. The proposed model makes use of a split-and-merge-mechanism to find story boundaries. The approach is based on visual features and text transcripts.

(iv) In Ma et al. [10], a set of key events are first detected from multimedia signal sources, including a large-scale concept ontology for images, text generated from automatic speech recognition systems, features extracted from audio track, and high-level video transcriptions. Then, a fusion scheme is investigated using the maximum figure-of-merit learning approach.

With the proposed method, we have obtained a recall of 0.878 and a precision of 0.767, which gives an $F1$ measure equal to 0.819 with a threshold optimized on the development set. On the same data set, our system is more effective than the actual systems. We have also tested our method using a feature vector expanded with a channel information (ABC or CNN), and the performance of the system was reached up to 0.825.

## 7. Conclusion

We have presented a method for segmenting TV news videos into stories. This system is based on multimodal features extraction. The originality of the approach is in the use of machine learning techniques for finding the candidate transitions from a large number of heterogeneous low-level features; it is also in the use of a temporal context for the features before their combination by early fusion.

This system has the advantage that it requires no or minimal external annotation. It was evaluated in the context of the TRECVID 2003 story segmentation task and obtained better performance than the current state of the art.

Future work would include other relevant descriptors for this task and an efficient step of normalization. Features of interest could be category topic detection using other sources in a video collection. Regarding the method for predicting the presence of story transition, it could be improved through a process that takes into account the video structure and the temporal information.

## Acknowledgment

This work was realized as part of the Quaero Programme funded by OSEO, French state agency for innovation.

## References

[1] A. F. Smeaton, P. Over, and W. Kraaij, "TRECVID—an overview," in *Proceedings of TRECVID*, 2003.

[2] T. S. Chua, S. F. Chang, L. Chaisorn, and W. Hsu, "Story boundary detection in large broadcast news video archives - Techniques, experience and trends," in *Proceedings of the 12th ACM International Conference on Multimedia*, pp. 656–659, October 2004.

[3] P. Joly, J. Benois-Pineau, E. Kijak, and G. Quénot, "The ARGOS campaign: evaluation of video analysis and indexing tools," *Signal Processing*, vol. 22, no. 7-8, pp. 705–717, 2007.

[4] A. E. Abduraman, S. A. Berrani, and B. Mérialdo, "TV program structuring techniques: a review," in *TV Content Analysis: Techniques and Applications*, 2011.

[5] J. M. Gauch, S. Gauch, S. Bouix, and X. Zhu, "Real time video scene detection and classification," *Information Processing and Management*, vol. 35, no. 3, pp. 381–400, 1999.

[6] L. Chaisorn and T. S. Chua, "Story boundary detection in news video using global rule induction technique," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '06)*, pp. 2101–2104, July 2006.

[7] L. Chaisorn, T. S. Chua, and C. H. Lee, "A multi-modal approach to story segmentation for news video," *World Wide Web*, vol. 6, no. 2, pp. 187–208, 2003.

[8] H. Misra, F. Hopfgartner, A. Goyal et al., "Tv news story segmentation based on semantic coherence and content similarity," in *Proceedings of the 16th international conference on Advances in Multimedia Modeling*, pp. 347–357, 2010.

[9] A. Goyal, P. Punitha, F. Hopfgartner, and J. M. Jose, "Split and merge based story segmentation in news videos," in *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, pp. 766–770, 2009.

[10] C. Ma, B. Byun, I. Kim, and C. H. Lee, "A detection-based approach to broadcast news video story segmentation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 1957–1960, April 2009.

[11] E. Dumont and B. Mérialdo, "Split-screen dynamically accelerated video summaries," in *Proceedings of the 1st TRECVID Video Summarization Workshop (TVS '07)*, pp. 55–59, September 2007.

[12] A. G. Hauptmann, M. G. Christel, W. H. Lin et al., "Clever clustering vs. simple speed-up for summarizing BBC rushes,"

in *Proceedings of the 1st TRECVID Video Summarization Workshop (TVS '07)*, pp. 20–24, September 2007.

[13] E. Dumont and B. Mérialdo, "Automatic evaluation method for rushes summary content," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '09)*, pp. 666–669, July 2009.

[14] C. Snoek, M. Worring, and A. W. M. Smeulders, "Early versus late fusion in semantic video analysis," in *Proceedings of the 13th Annual ACM International Conference on Multimedia*, pp. 399–402, 2005.

[15] M. A. Hearst, "Multi-paragraph segmentation of expository text," in *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics (ACL '94)*, pp. 9–16, 1994.

[16] G. Quénot, D. Moraru, and L. Besacier, "CLIPS at TRECvid: shot boundary detection and feature detection," in *Proceedings of TRECVID*, 2003.

[17] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–38, 1998.

[18] J. Poignant, L. Besacier, G. Quénot, and F. Thollard, "From text detection in videos to person identification," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2012.

[19] J. L. Gauvain, L. Lamel, and G. Adda, "The LIMSI broadcast news transcription system," *Speech Communication*, vol. 37, no. 1-2, pp. 89–108, 2002.

[20] V. B. Le, O. Mella, and D. Fohr, "Speaker diarization using normalized cross likelihood ratio," in *Proceedings of the 8th Annual Conference of the International Speech Communication Association (Interspeech '07)*, pp. 873–876, August 2007.

[21] M. A. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *SIGKDD Explorations Newsletter*, vol. 11, pp. 10–18, 2009.

[22] http://www.cs.waikato.ac.nz/ml/weka/.