Hindawi Publishing Corporation
International Journal of Digital Multimedia Broadcasting
Volume 2010, Article ID 709161, 11 pages
doi:10.1155/2010/709161

Research Article

Statistical Skimming of Feature Films

Sergio Benini, Pierangelo Migliorati, and Riccardo Leonardi

Department of Information Engineering DII - SCL, Università di Brescia, via Branze 38, 25123 Brescia, Italy

Correspondence should be addressed to Sergio Benini, sergio.benini@ing.unibs.it

Received 29 August 2009; Accepted 21 December 2009

Academic Editor: Ling Shao

We present a statistical framework based on Hidden Markov Models *(HMMs)* for skimming feature films. A chain of *HMMs* is used to model subsequent story units: *HMM* states represent different visual-concepts, transitions model the temporal dependencies in each story unit, and stochastic observations are given by single shots. The skim is generated as an observation sequence, where, in order to privilege more informative segments for entering the skim, shots are assigned higher probability of observation if endowed with salient features related to specific film genres. The effectiveness of the method is demonstrated by skimming the first thirty minutes of a wide set of action and dramatic movies, in order to create previews for users useful for assessing whether they would like to see that movie or not, but without revealing the movie central part and plot details. Results are evaluated and compared through extensive user tests in terms of metrics that estimate the content representational value of the obtained video skims and their utility for assessing the user's interest in the observed movie.

"*I took a speed reading course and read "War and Peace" in 20 minutes. It involves Russia.*"

Woody Allen.

1. Introduction

In the last years, with the proliferation of digital TV broadcasting, dedicated internet websites, and private recording of home video, a large amount of video information has been made available to end-users. Nevertheless, this massive proliferation in the *availability* of digital video has not been accompanied by a parallel increase in its *accessibility*. In this scenario, video summarization techniques may represent a key component of a practical video-content management system. By watching a condensed video, a viewer may be able to assess the relevance of a programme before committing time, thus facilitating typical tasks such as browsing, organizing, and searching video-content.

For unscripted-content videos such as sports and home-videos, where the events happen spontaneously and not according to a given script, previous work on video summarisation mainly focused on the extraction of *highlights*. Regarding scripted-content videos—those videos which are produced according to a script, such as feature films (e.g., Hollywood movies), news and cartoons—two types of video abstracts have been investigated so far, namely, *video static summarization* and *video skimming*. The first one is a process that selects a set of salient key-frames to represent content in a compact form and present it to the user as a static programme preview. Video skimming instead, also known as *video dynamic summarization*, tries to condense the original video in the more appealing form of a shorter video clip. The generation of a skim can be viewed as the process of selecting and gluing together proper video segments under some user-defined *constraints* and according to given *criteria*. On the one hand, final user *constraints* are usually defined by the time committed by the user to watch the skim, which in the end determines the final skim ratio. On the other hand, skimming *criteria* used to select video segments range from the use of motion information [1], the exploitation of the hierarchical organization of video in scenes and shots as in [2], or the insertion of audio, visual, and text markers [3].

In this paper, in order to derive the skim, we propose to combine the information deriving from the story structure with the characterization of the shots in terms of salient features, that are motion dynamics for action movies, and the presence of human faces for dramas, respectively. These salient features inherently estimate the contribution

of each shot in terms of "content informativeness" and determines whether the shot will be included in the final skim. The "structure informativeness" of the video is instead captured by *HMMs*, which model semantic scenes and whose observations produce the shot sequence of the final skim.

The paper is organized as follows. Section 2 gives a brief overview on the current state-of-the-art related to video skimming and the other techniques here employed. Section 3 presents the criteria adopted to realize the skim. In Section 4, we characterize the content informativeness of shots by the use of salient features related to the movie genres. Section 5 describes how to model each story unit by an *HMM*. In Sections 6 and 7, the video skims are generated and evaluated, while in Section 8 conclusions are drawn.

2. Related Work

In the past, *HMM* has been successfully applied to different domains such as speech recognition [4], genome sequence analysis [5], and so forth. For video analysis, *HMMs* have been used to distinguish different genres [6], and to delineate high-level structures of soccer games [7]. In this work instead, *HMMs* are used as a unified statistical framework to represent visual-concepts and to model the temporal dependencies in story units with the aim of video skimming.

Even if the interest in effective techniques for dynamic video skimming is highly in demand, to date, there are relatively less works that address dynamic video skimming than works related to static video summarisation (see, e.g., [8] for a systematic classification of previous works on condensed representations of video content). In general, to process huge quantities of video frames is more difficult than to select a subset of relevant key-frames. It is also challenging to define which segments have to be highlighted and mapping the mechanisms of human perception into an automated abstraction process. For these reasons, at the moment most current video skimmings are intended as natural evolutions of the methods employed for generating the related static summaries. Therefore, many skimming methodologies rely on the same clustering algorithms which have been adopted to obtain static video summaries, such as those that have been extensively reviewed in [9]. For example, in one of the latest works [10], the authors propose an algorithm for video summarization which first constructs story boards and then it removes redundant video content using hierarchical agglomerative clustering at the key-frame level.

Since it is easy to understand how a tool that can automatically shorten the original video while preserving only the important content would be greatly useful to most users, alternative skimming methods have been developed in time. The oldest and most straightforward approach is to compress the original video by speeding up the playback without considerable distortion, as pointed out by Omoigui [11]. A similar approach is also described in [12], where an audio time-scale modification scheme is applied. However, these techniques only allow a maximum time compression of around 2.5 times, depending on the rate of speech; since once the compression factor goes beyond this range, the perceived speech quality becomes quite poor or annoying. A similar method is found in [13], where the skim generation is formulated as a rate-distortion optimisation problem.

A number of approaches use attention and saliency models to derive the video skim. In [14] summaries are generated by merging together those video segments that contain high-confidence scores in terms of motion-attention. In a further generalisation described in [1], the same authors take into account also the presence of human faces and the present audio information. One limitation of these two approaches, that we try to overcome in this work, is that the structural information such as the intershot relationship is not exploited for video skimming. As a result, the produced dynamic video summary is purely a collection of video highlights in terms of attention model and does not take into account the content coverage and relationships. In [15] a method for the detection of perceptually important video events, based on saliency models for the audio, visual and textual information, is also described.

Other techniques rely on the presence of textual information only. The Informedia project [3], for example, concatenates audio and video segments that contain preextracted text key-words to form the skim, for example from news.

Without relying on text cues, in [16] skims are generated by a dynamic sampling scheme. Videos are first decomposed into subshots, and each subshot is assigned a motion-intensity index. Key-frames are then sampled from each subshot based on an assigned rate which is derived by the motion index. During the skim playback, techniques of linear interpolation are adopted to provide users with a dynamic storyboard. Similar methods for skim generation based on precomputed key-frames are described in [17, 18].

Singular Value Decomposition and Principal Component Analysis have been also proposed in [19] as attractive models for video skimming. However, these techniques remain computationally intensive since they process all video frames, which cannot be practical for huge repositories.

More recently, research on generating skims for specific unscript-content video has been reported. For example, ad-hoc summarisation methods for rushes were designed within the RUSHES project [20] and have been a field of competition in the TrecVid 2008 [21] as in the related work in [22].

In [23], a skimming system for news is presented. By exploiting news content structure, commercials are removed by using audio cues, and then anchor persons are detected (using a Gaussian Mixture Model) and glued together to form the skim.

Home video skimming is addressed in many works, such as in [2, 24]. In this last work, video skimming is based on media aesthetics. Given a video and a background music, this system generates a music-video-style skimming video automatically, with consideration of video quality, music tempo, and the editing theory.

Some research efforts [25] have been investigating the generation of skims for sports videos based on the identification of exciting highlights such as soccer goals or football touchdowns, therefore, according to the significance of play scenes.

Finally, regarding movies and narrative video, the rules of cinematic production are exploited in [26–28] to produce a syntactical-based reduction scheme for skim generation. Based on both audio and visual information, some utility functions are modelled to maximise the content and coherence of the summaries.

3. Skimming Criteria

Since a skimming application should automatically shorten the original video while preserving the important and informative content, in this work it is proposed that the time allocation policy for realising a skim should fulfil the following criteria.

(i) "*Coverage*". The skim should include all the parts of the movie structure into the synopsis. Since in the movie cinematic syntax, the story structure constitutes a fundamental element for conveying the movie message, each *Logical Story Units (LSU)*, that is, each "sequence of contiguous and interconnected shots sharing a common semantic thread" [29] (which is the best computable approximation to a semantic scene), should participate the skim. Therefore, if V is the original video of total length $l(V)$, we consider it to be already segmented into n Logical Story Units Λ_i by previous analysis as in [30] or in [31], that is, $V = \{\Lambda_1, \Lambda_2, \ldots, \Lambda_n\}$. The final skim v will contain the skimmed version of each *LSU*, that is, $v = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$.

(ii) "*Representativeness*". Each Logical Story Unit should be represented in the skim proportionally to its duration in the original video. Therefore, if r is the skimming ratio defined by the user (i.e., $r = l(v)/l(V)$), the length of each *LSU* in the synopsis should be $l(\lambda_i) = r \cdot l(\Lambda_i)$, for all $i = 1, \ldots, n$.

(iii) "*Structure Informativeness*". Since in the movie cinematic syntax, the information which is introduced by the film editing process, especially by the shot patterns inside story units (e.g., dialogues, progressive scenes, etc.), is relevant for the storytelling and for conveying the plot, this information should be preserved in the final skim. Therefore, if Λ_i is a *dialogue* in the movie V, the corresponding story unit $\lambda_i \in v$ should preserve the dialogue structure.

(iv) "*Content Informativeness*". To represent each story unit, the most "informative" video segments should be preferred. Of course, "informative" is a term that can assume multiple meanings depending on context. Since we are dealing with movies, a segment is intended as "informative" if it is effective in conveying the general concept presented in the film. One possibility of relating this to some physical properties of the video material is to link the concept of "informative" segment with the genre of the movie. If the film is an action movie, for example, an informative segment will be a high dynamic one, while in case of a dramatic film, informative segment will be those showing key dialogues between main characters. We can therefore quantify the "informativeness" of a video segment by assessing in the video the presence of one *salient feature* $\mathcal{F}$ which is related to the film genre.

4. Salient Features

In order to assess the *content informativeness* of each shot of the film, we introduce the concept of salient feature $\mathcal{F}$ related to the movie genre. The skimming procedure that follows this description is general and can be applied to any film provided that a salient feature for that movie genre is defined. The user can also choose to apply a salient feature which is not related to the movie genre, just because he/she is more interested in that, or as a leisure activity, for example, in the context of a video mash-up application. There is of course no limit to the set of salient features that can be defined.

In order to provide a couple of examples of possible salient features, we shortly describe in the following a measure of motion activity which can be useful to skim action movies, and a face detection procedure to assess the presence of human faces in dramatic movies, where most information is conveyed by dialogues between characters.

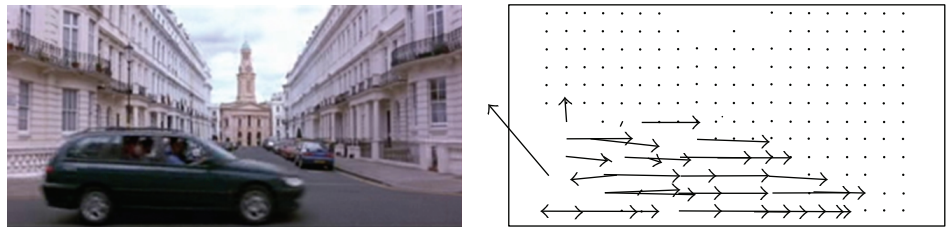
4.1. Motion Activity. The intensity of motion activity is a subjective measure of the perceived intensity of motion in a video segment. For instance, while an "*anchorman*" shot in a news program is perceived by most people as a "low-intensity" action, a "*car chasing*" sequence would be viewed by most viewers as a "high-intensity" sequence.

As stated in [32], the intensity of motion activity in a video segment is in fact a measure of "how much" the content of a video is changing. Motion activity can be therefore interpreted as a measure of the "entropy" (in a wide sense) of a video segment. We characterize the motion activity of video shots by extracting the motion vector *(MV)* field of *P*-frames (see Figure 1) directly from the compressed *MPEG* stream, thus allowing low computational cost.

For compression efficiency, *MPEG* uses a motion-compensated prediction scheme to exploit temporal redundancy inherent in an image sequence. In each *GOP* (Group of Pictures), *I*-frames are used as references for the prediction. *P*-frames are coded using motion-compensated prediction from a previous *P* or *I*-frame (forward prediction), while *B*-frames are coded by using past and/or future pictures as references. This means that, in order to reduce the bit-rate, macroblocks (*MBs*) in *P* and *B*-frames are coded using their differences with corresponding reference *MBs*, and a motion vector carries the displacement of the current *MB* with respect to a reference *MB*.

The raw *MV* field extracted turns out to be normally rough and erratic, and not suitable for tasks such as accurately segmenting moving objects. However, after being properly filtered, the *MVs* can be very useful to describe the general motion dynamics of a sequence, thus characterising the amount of visual information conveyed by the shot.

The filtering process applied includes first removing the *MVs* next to image borders which tend to be unreliable, then using a texture filter, followed by a median filter. The texture filter is needed since, in the case of low-textured uniform



(a) (b)

Figure 1: A decoded *P*-frame and its motion vector field.

areas, the correlation methods used to estimate motion often produce spurious *MVs*. After having filtered the motion vectors on texture criterion, a median filtering is used to straighten up single spurious vectors such as those that could still be present close to borders.

In general, the perceived motion activity in a video is higher when the objects in the scene move faster. In this case the magnitudes of the *MVs* of the macroblocks (*MBs*) that make up the objects are significant, and one simple measure of motion intensity can be extracted from the *P*-frame by computing the mean μ_P of the magnitudes of motion vectors belonging to intercoded *MBs* only.

However, most of the perceived intensity in a video is due to objects which do not move according to the uniform motion of the video camera. Thus, a good *P*-frame-based measure of motion intensity is given by the standard deviation σ_P of the magnitudes of motion vectors belonging to intercoded *MBs*.

The measure σ_P can be also extended to characterize the motion intensity $\mathcal{MI}(S)$ of a shot *S*, by averaging the measures obtained on all the *P*-frames belonging to that shot. *MPEG7 Motion Activity* descriptor [32] is also based on a quantized version of the standard deviation of *MVs* magnitudes. For our purposes, each shot *S* is assigned its motion intensity value $\mathcal{MI}(S)$ in its not-quantized version. This value $\mathcal{MI}(S)$ tries to capture the human perception of the "intensity of action" or the "pace" of a video segment, by considering the overall intensity of motion activity in the shot itself (without distinguishing between the camera motion and the motion of the objects present in the scene). Since this is in fact a measure of "how much" the content of a video segment is changing, it can be interpreted as a measure of the "entropy" of the video segment, and can be used as a salient feature $\mathcal{F}$ for summarization purposes.

4.2. Presence of Human Faces. Since a user can be interested in privileging in the final skim the presence of shots containing human faces, for example, for visualising excerpts from dramatic movies, it is possible to define, for each shot, the salient feature $\mathcal{F}$ as the percentage of frames in the shot which contain at least one human face, subjected to a minimal dimension. In the actual implementation, the work by Viola and Jones described in [33] has been preferred to other face detection methods, but no restriction to other procedures is imposed. A possible extension of this salient feature related to dramatic movies is the integration of the information related to the human presence with the information related to the type of audio, for example, by detecting speech during dialogues.

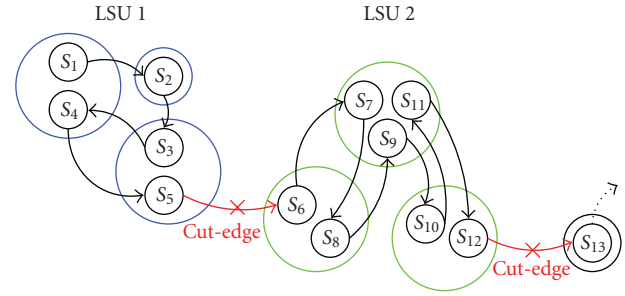


Figure 2: Detection of cut edges in a Scene Transition Graph. Since each cut edges is a sort of one-way transition from a highly connected group of clusters to another group, this can be considered as a reliable *LSU* boundary.

5. Modelling LSU with Hidden Markov Models

5.1. Logical Story Units Structure. In [30] it is shown how a video can be represented by a *Scene Transition Graph (STG)*, whose nodes are clusters of visually similar and temporally close shots, while edges between nodes stand for the transitions between subsequent shots. In the same work, the authors demonstrate that after the removal of cut-edges, that is, the edges which, if removed, lead to the decomposition of the *STG* into two disconnected subgraphs, each well connected subgraph represents a *Logical Story Unit (LSU)*, as shown in Figure 2.

In fact, since cut edges are one-way transitions from one set of clusters which are highly connected among each other (i.e., nodes connected by cycles in the corresponding indirect graph, see [30]) to another set of clusters characterized by a completely new visual-content, cut edges can be then considered as reliable *LSU* boundaries.

The *STG* has been computed on the base of an *LSU* segmentation obtained as in [31]. On the shot level, any existing technique for shot boundary detection can be employed, without loss of generality. However, the algorithm here employed adopts the classical twin comparison method, where the error signal used to detect transitions is based on statistical modeling [34].

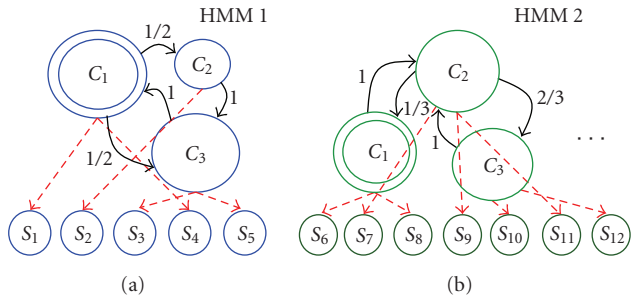


FIGURE 3: *LSUs* of Figure 2 are equivalently modeled by *HMMs*. States $\{C_i\}$ correspond to distinct nodes of the *STG* subgraph; transition probabilities are computed according to the relative frequency of the transitions between clusters, and shots $\{S_i\}$ are the possible observation set. The *HMM* initial states have been indicated with a double circle.

Starting from the *STG* representation, each *LSU* can be equivalently modeled by an *HMM*. This is a discrete state-space stochastic model which works well for temporally correlated data streams, where the observations are a probabilistic function of a hidden state [4]. Such a modelling choice is supported by the following considerations [7]:

(1) Video structure can be described as a discrete state-space, where each state is a conveyed *concept* (e.g., "*man face*") and each state-transition is given by a change of concept;

(2) The *observations* of concepts are stochastic since video segments seldom have identical raw features even if they represent the same concept (e.g., more shots showing the same "*man face*" from slightly different angles);

(3) The sequence of concepts is highly correlated in time, especially for scripted-content videos (movies, etc.) due to the presence of editing effects and typical shot patterns inside scenes (i.e., dialogues, progressive scenes, etc.).

For our aims, *HMM* states representing concepts will correspond to distinct clusters of visually similar shots (where clusters are obtained as described in [9]); state transition probability distribution will capture the shot pattern structure of the *LSU*, and shots will constitute the observation set (as shown in Figure 3).

5.2. HMM Definition. We now define how the *HMM* is built, and then how the models generate observation sequences in order to produce the video skim. Formally, an *HMM* representing an *LSU* is specified by the following.

(i) N, the Number of States. Although the states are hidden, in practical applications there is often some physical significance associated to the states. In this case, we define that each state corresponds to a distinct node of an *STG* subgraph: each state is one of the N clusters of the *LSU* containing a number of visually similar and temporally close shots. We denote states as $C = \{C_1, C_2, \ldots, C_N\}$, and the state at time t as q_t.

(ii) M, the Number of Distinct Observation Symbols. The observation symbols correspond to the output of the system being modeled. In this case, each observation symbol $S = \{S_1, S_2, \ldots, S_M\}$ is one of the M shots of the video.

(iii) $\Delta = \{\delta_{ij}\}$, *the State Transition Probability Distribution:*

$$\delta_{ij} = P\left[q_{t+1} = C_j \mid q_t = C_i\right], \quad 1 \le i,\ j \le N. \tag{1}$$

Transition probabilities are computed as the relative frequency of transitions between clusters in the *STG*, that is, δ_{ij} is given by the ratio of the number of edges going from cluster C_i to C_j to the total number of edges departing from C_i. In a *HMM*, states can be interconnected in such a way that any state can be reached from any other state (e.g., an ergodic model); for this special case, we would have $\delta_{ij} > 0$ for all (i, j). However, since in our case the interconnections of states are given by the transitions from shot to shot, and not all clusters are interconnected with all the others, this usually makes the model a nonergodic one; in this case it is likely that we have $\delta_{ij} = 0$ for one or more (i, j) pairs.

(iv) $\Sigma = \{\sigma_j(k)\}$, *the Observation Symbol Distribution,* where

$$\sigma_j(k) = P\left[S_k \text{ at } t \mid q_t = C_j\right], \quad 1 \le j \le N,\ 1 \le k \le M \tag{2}$$

We define the observation symbol probability in state C_j, that is, $\sigma_j(k)$, as the ratio of the salient feature value in the shot S_k to the total value of the salient feature of the cluster that contains S_k. It represents the probability for the shot S_k of being chosen as observation of the related visual concept, and it is defined as

$$\sigma_j(k) = \begin{cases} \dfrac{\mathcal{F}(S_k)}{\mathcal{F}\left(C_j\right)} & \text{if } S_k \in C_j \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where $\mathcal{F}(C_j)$ is defined as the sum of all the salient feature values of the shots belonging to cluster C_j, that is, $\mathcal{F}(C_j) = \sum_{S_h \in C_j} \mathcal{F}(S_h)$. Conversely, if the shot does not belong to the cluster, its observation probability is null, so that it cannot be selected to represent the cluster visual concept.

(v) $\pi = \{\pi_i\}$, *the Initial State Distribution,* where

$$\pi_i = P[q_1 = C_i], \quad 1 \le 1 \le N. \tag{4}$$

In order to preserve the information about the entry point of each *LSU*, $\pi_i = 1$ if the cluster C_i contains the first shot of the *LSU*, otherwise $\pi_i = 0$.

Therefore, a complete specification of the *HMM* requires two model parameters (N and M), the observation symbols S, and the probability distributions Δ, Σ, and π. Since the set $S = \{S_1, S_2, \ldots, S_M\}$ is common to all the *HMMs*, for convenience, we can use the compact notation $\Gamma = (\Delta, \Sigma, \pi, N)$ to indicate the complete parameter set of the *HMM* representing an *LSU*.

6. Stochastic Skim Generation

In order to generate an informative skim, the following solutions have been adopted to fulfill all the skimming criteria stated in Section 3.

(i) Coverage. Since the skim should include all the semantically important story units, each detected *LSU* participates to the final synopsis. As a general remark, please notice that the skim ratio r should be subject to a minimal value $r_{\min}$ for the skim to be representative of the movie structure and content.

(ii) Representativeness. Let $l(\Lambda_1), l(\Lambda_2), \ldots, l(\Lambda_n)$ be the lengths of the n *LSUs* that compose the original video. Then in the skim, for each *LSU*, a time slot of length $l(\lambda_i)$ is reserved, where

$$l(\lambda_i) = r \cdot l(\Lambda_i), \quad \forall i = 1, \ldots, n. \tag{5}$$

(iii) Structure Informativeness. In order to include in the synopsis the information conveyed by the shot patterns inside the story units, the following procedure is adopted. Since the state transition probability distribution of *HMM* Γ_i has statistically captured the structure of the transitions between shots inside the corresponding *LSU* Λ_i, a skimmed version λ_i of the *LSU* can be generated as an observation sequence of the associated *HMM*, Λ_i, that is:

$$\lambda_i = O_1 O_2 \cdots, \tag{6}$$

where each observation O_j is one of the symbols from S, that is, a shot of the original video.

Starting from the first Hidden Markov Model Γ_1, the sequence λ_i is generated as follows:

(1) choose the initial state $q_1 = C_h$ according to the initial state distribution π. Set $t = 1$;

(2) while (total length of already concatenated shots) < (time slot $l(\lambda_i)$ assigned to the current *LSU*),

   (a) choose $O_t = S_k$ according to the symbol probability distribution in state C_h, that is, $\sigma_h(k)$;

   (b) transit to a new state $q_{t+1} = C_j$, according to the state transition probability for state C_h, that is, δ_{hj};

   (c) set $t = t + 1$;

(3) repeat the previous steps for all Γ_i.

The above described procedure means that the generated skim for each *LSU* is one of the possible realizations of the stochastic process described by the corresponding *HMM*, where both interstate transitions and the shot selections are the results of random trials. In order to generate the whole skim, this method is applied to all Logical Story Units, and the obtained sequences of observed shots for each *LSU* are concatenated in the final synopsis.

Since the skim generation does not take into account the original shot order inside a story unit, it may happen that in the skim a shot which is later in the original *LSU* can appear before another one which is actually prior to it (as it sometimes happens in commercial trailers). In the circumstance of "anticasual" shots, they are repositioned in causal order inside each *LSU*, without altering the nature of the algorithm. The shot repositioning is automatically performed on the basis of the shot identifiers (corresponding to the shot positions in the movie), while shots belonging to different *LSUs* are already in casual order by construction.

(iv) Content Informativeness. In order to privilege more "informative" shots, the observation symbol probability distribution Σ depends on the presence of the salient feature. In particular, the higher is the value of the salient feature in a shot S_k of the cluster C_j, the higher will be $\sigma_j(k)$, that is, S_k will be more likely chosen for the skim.

For example, regarding action movies and the salient feature related to motion activity, by assigning higher probability of observation to more dynamic shots, we privilege "informative" segments for the skim generation. At the same time, we avoid to discard *a-priori* low-motion shots, that can be chosen as well for entering the skim, even if with lower probability. Moreover, once that one shot is chosen for the video skim, it is removed from the list of candidates for further time slots, and the observation symbol distribution is recomputed on remaining shots in the cluster. This prevents the same shot from repetitively appearing in the same synopsis, and at the same time allows also low-motion shots to enter the skim, if the user-defined skim ratio is large enough. Therefore, as it should be natural, in very short skims, "informative" shots are likely to appear first, while for longer skims, even less "informative" shots can enter the skim later on.

7. User Tests and Performance Evaluation

To quantitatively investigate the performance of the proposed method for video skimming, we carried out two main experiments using the feature films in Tables 1 and 3.

7.1. User Test A: Informativeness and Enjoyability. In this first test, for the evaluation of the skims, the method and the criteria of "*informativeness*" and "*enjoyability*" adopted in [2] have been used. *Informativeness* assesses the capability of the statistical model of maintaining content, coverage, representativeness, and structure, while reducing redundancy. *Enjoyability* assesses the performance of the salient feature employed in selecting perceptually enjoyable video segments for the skim. Starting from the *LSU* segmentation results, as generated in [31], we produced 20 dynamic summaries with their related soundtracks: for each video in Table 1, two associated skims have been produced, one with 10% of the original video length and the other with the 25%. The salient feature related to motion activity was used for 5 movies (no. 1, no. 4, no. 7, no. 8, no. 9) which are closer to the genre

Table 1: Set of feature films (TEST A).

No.	Video
1	A Portoguese farewell
2	Notting Hill
3	A beautiful mind
4	Pulp fiction
5	Camilo and Filho
6	Riscos
7	Altrimenti ci arrabbiamo
8	Don Quixotte
9	Più forte ragazzi
10	Don Camillo

"action movie", while the presence of faces was adopted for the other ones closer to the genre "drama".

A first set of 12 students (6 male, 6 female) assessed the quality of the produced skims by watching, for each movie, one randomly selected version among the available three: 10%, 25%, and the original movie 100%. Before starting the test, the participants were given a short oral introduction about the idea of automatic video skim generation and on the purpose of the test. After watching the selected version, each student assigned two scores ranging from 0 to 100, in terms of *informativeness* and *enjoyability*, also in case they watched the original movie if they thought that this was not 100% enjoyable or informative (e.g., when an intricate plot determines that the movie is not completely informative regarding situations and displayed events).

Table 2 shows the obtained average scores which have been normalized by the score assigned to the original movie.

In these experiments, average normalized scores for *enjoyability* are around 72% and 80% for video skims of 10% and 25% length, respectively. Regarding *informativeness*, average normalized scores are around 69% and 81%, respectively. These results are comparable with the ones presented in one of the most referenced works on video skims [2]. However, results presented here have been obtained on a larger set of videos, in particular on movies coming from different genres.

7.2. User Test B: Utility and Comparison. Based on the user test A only, it is not possible to completely assess the utility of the generated skim, nor to evaluate the algorithm performance with respect to other solutions. For this reason, another user set of 12 students (6 male, 6 female) were hired for performing a more severe test on another set of 10 movies (in Table 3) concerning the skim utility in a modern multimedia management system.

It is nowadays believed by broadcasters and content producers that a skim of a movie would be helpful for the user to assess whether he/she would be interested in paying to watch the entire movie. Of course the skim should not reveal too much about the movie plot, for example, being limited only to the introductory part of the film. A collection of movie skims could be offered as a preview on websites to be watched by users so that they can decide whether or not to download the whole movie.

Table 2: Performance evaluation of Video Skimming.

No.	Enjoyability			Informativeness		
	10%	25%	100%	10%	25%	100%
1	69.3	75.8	91.9	61.8	72.1	90.3
	75.4	**82.4**	**100**	**68.4**	**79.8**	**100**
2	62.8	70.5	86.2	65.4	75.8	93.1
	72.8	**81.8**	**100**	**70.3**	**81.4**	**100**
3	68.2	71.4	88.2	65.6	78.8	89.4
	77.2	**80.9**	**100**	**73.4**	**88.1**	**100**
4	57.5	67.2	84.6	63.3	72.9	91.5
	68.0	**79.4**	**100**	**69.1**	**79.6**	**100**
5	68.1	73.6	94.2	65.1	72.3	93.4
	72.3	**78.1**	**100**	**69.7**	**77.4**	**100**
6	55.2	68.8	93.0	64.0	78.1	94.0
	59.3	**73.9**	**100**	**68.1**	**83.0**	**100**
7	66.5	78.4	91.2	60.3	78.0	93.8
	72.9	**85.9**	**100**	**64.3**	**83.2**	**100**
8	69.5	74.9	90.5	60.2	72.1	89.5
	76.8	**82.7**	**100**	**67.2**	**80.5**	**100**
9	69.8	70.1	85.5	62.8	72.1	92.1
	81.6	**82.0**	**100**	**68.1**	**78.2**	**100**
10	65.8	68.1	95.3	65.6	71.2	93.0
	69.0	**71.4**	**100**	**70.5**	**76.5**	**100**
Aver.	72.5	79.8	—	68.9	80.8	—
Drop	27.5	20.2	—	31.1	19.2	—

With this scenario in mind, the user tests were performed according to the following procedure. From the introductory part (i.e., the first 30 minutes) of the 10 blockbuster movies in Table 3, three skims have been generated by three different methods with the same skim ratio $r = 0.1$, so that each skim is about 3 minutes long.

The participants were told that the video skims were automatically generated by algorithms and that the aim of the experiments was a comparison of three automatic video skim generation algorithms. After answering three questions about gender, age, and film watching behaviour, each user was requested to watch 10 randomly chosen skims, one per movie, without being aware of which algorithm was responsible for the creation of a particular skim.

The first adopted algorithm $\mathcal{A}$ generates a video skim by selecting video segments randomly from the original movie. The second algorithm $\mathcal{B}$ is the algorithm described in this work, based on the use of salient features and Hidden Markov Models, where *LSUs* were generated as in [31] and the salient feature related to motion activity was used for action movies no. 1, no. 2, no. 5, no. 9, and no. 10, while the presence of faces was adopted for the other ones closer to the genre "drama", according to their IMBd classification [35]. In Table 3, more details can be found about the movie shots, the number of *LSUs*, the skim length, and the number of visual concepts (i.e., the hidden states) used to generate the skim according to the proposed method.

Table 3: Set of feature films (TEST B) and details about the structure of movies and skims.

No.	Video	LSUs	Shots (30 min)	Shots (3 min)	Visual concepts
1	Raiders of the lost ark	13	353	35	91
2	Terminator	13	399	37	117
3	Gattaca	30	246	14	85
4	Donnie Darko	12	256	25	31
5	Finding Nemo	14	462	31	80
6	A Beautiful mind	15	353	40	39
7	Talk to her	10	168	24	41
8	Goodbye Lenin	20	499	47	91
9	Kill Bill vol 2	7	224	33	15
10	War of the worlds	15	280	30	74

Table 4: Questionnaire about the utility and quality of watched skims.

No.	Question
1	Is the skim useful for understanding the genre of the original movie? (1–5)
2	Is the skim able to give you a clear idea of the movie atmosphere? (1–5)
3	Is the skim able to give you a clear idea of the narration pace? (1–5)
4	Is the skim able to give you a clear idea about the involved characters? (1–5)
5	Is the skim useful for understanding whether you would/would not like to watch the entire movie? (1–5)
6	Please give the skim a global score. (1–5)

The third set of skims, generated with method $\mathcal{C}$, have been manually generated by a cinema lover and expert of editing systems, aiming at providing an overview of the storyline and a fair impression of the movie atmosphere.

We expect that skims generated using the *random* technique would be of lower quality than skims generated by our *HMM* approach. On the other side, *manually made* skims certainly represent an upper-bound for the overall quality of skims. Therefore, we assume that in terms of quality of results, the *random* method will be worse than the *HMM* approach, and *manually made* skims will have the highest possible quality.

After viewing each skim, participants were requested to fill out a questionnaire (as in Table 4) about the utility and the quality of the watched skim, and to mark each of the 6 questions on a Likert scale from 1 (min) to 5 (max). After answering, participants were also asked for detailed comments and whether they had already seen the original movie within the last 6 months, more than 6 months ago, only partially or never before, in order to accordingly weight their answers.

Results regarding the quality of the three compared skims are reported in Figure 4. It is evident that for all questions, the *manually made* skims stand out as better with respect to *HMM* and random. As expected, the *HMM* skims score on average better than the *random* method.

Table 5: Questionnaire about the informativeness of "Raiders of the lost ark".

No.	Question
1	Why does the room in the cave collapse?
2	What the main male character escape with?
3	What is the topic of the talk in the library?
4	What is the location of the last scene?

The informativeness of each skim was also investigated by asking the users 4 specific questions regarding the plot understanding of the original movie that can be inferred by watching the skimmed version. The proposed questions concern four main narrative key-points (judged by a human) which take place in the considered first 30 minutes of each movie (see, e.g., Table 5 with the questions related to the introductory part of the movie "Raiders of the lost ark").

Marks from 0 to 2 were given to wrong/missing, partially correct, and correct answers, respectively. Answers from users that declared that they have seen the movie before have been weighted accordingly.

Results regarding the informativeness of the three compared skims are reported in Figure 5. It is evident that for all questions regarding the level of understanding of the plot, the *manually made* skims stand out as better with respect to *HMM* and random. As expected, the *HMM* skims score on average better than the *random* approach.

Regarding the proposed methodology (algorithm $\mathcal{B}$), we consider $r = 0.1$ as the lowest skimming ratio that we can applied to a movie before producing a degenerate skim, that is, no more representative of the movie structure and content. For smaller values of r, in fact, the structure of some scenes would be completely lost. Therefore, we have produced our experiments testing the system in limit conditions, and we expect performance to be even better when bigger skimming ratios are applied.

Further analysis and discussion on obtained results are ongoing to critically revise results obtained for movies no. 4 and no. 10 whose results for both experiments in test B are not aligned with the rest of the films. In particular we plan to carefully analyse shooting scripts since we guess that,

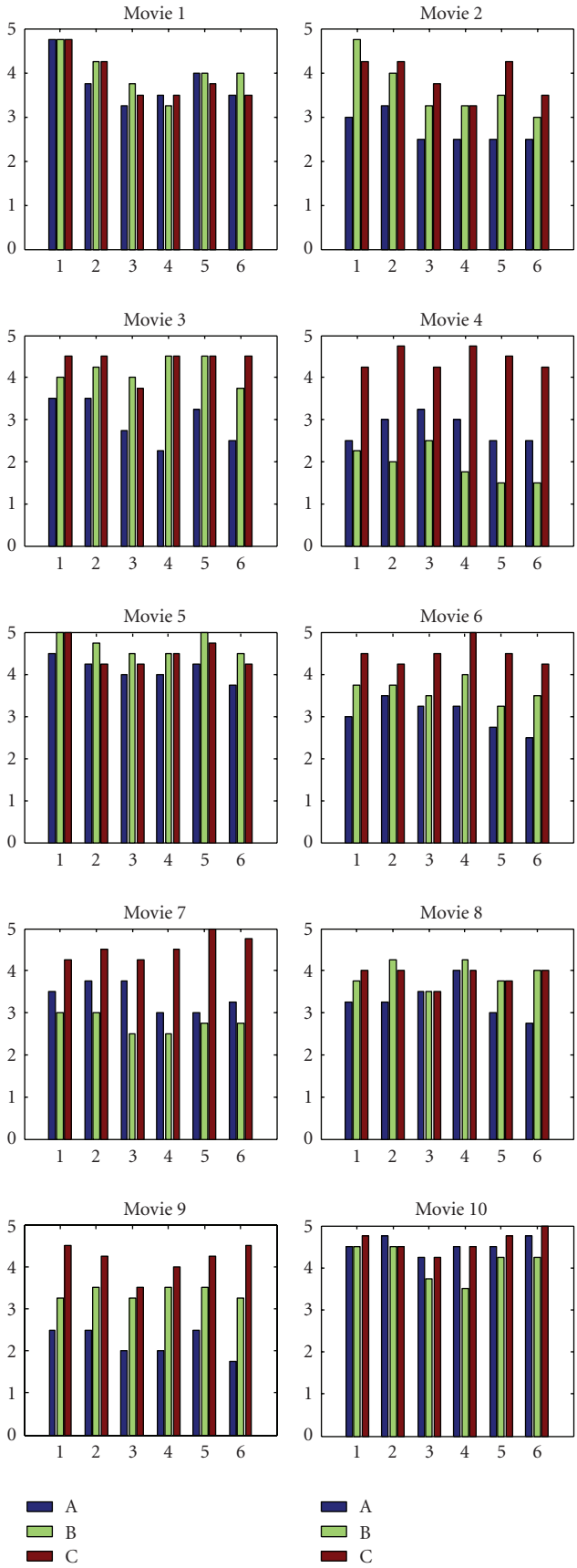


Figure 4: Average marks on skim quality (A = random sampling) (B = HMM and salient features) (C = manual).

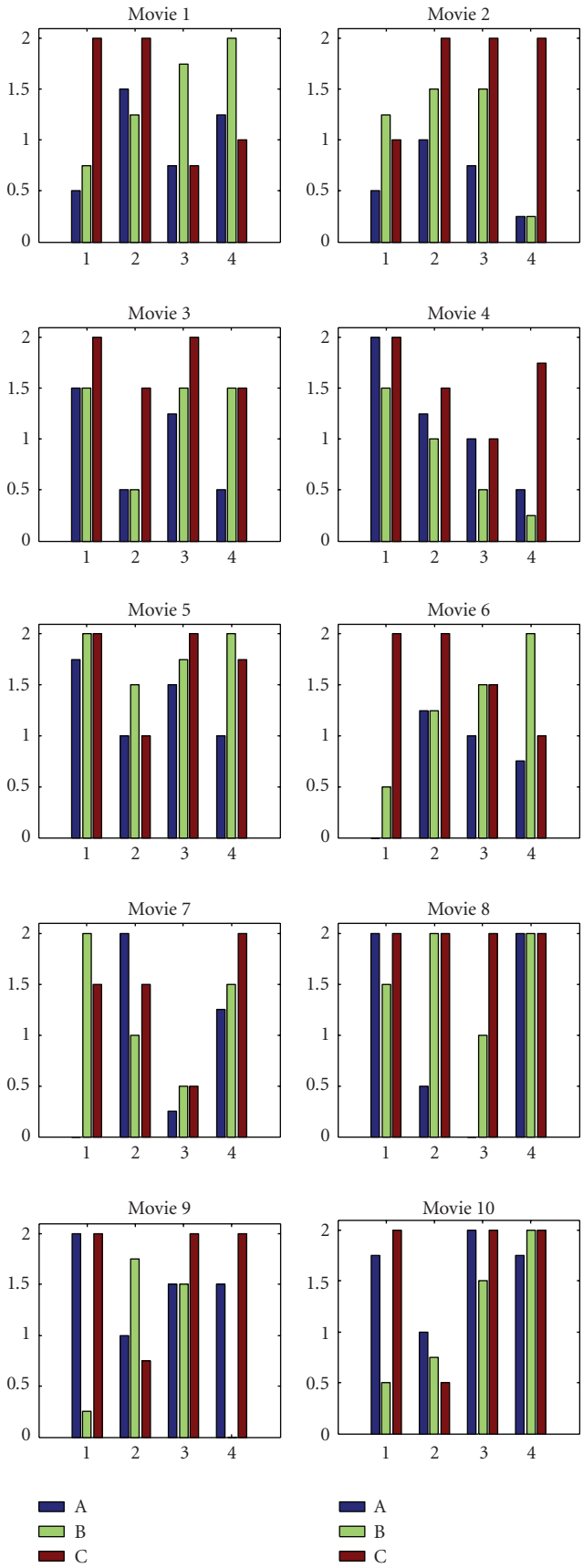


Figure 5: Average marks on plot understanding (A = random sampling) (B = HMM and salient features) (C = manual).

for both movies, misaligned results are probably due to the intricate plots (e.g., the use of flashbacks, the mixing of real scenes with ones taken from dreams, etc.) and the peculiar editing styles employed by both directors.

8. Conclusions

In this paper a method for video skim generation has been proposed. This technique is based on a previous high-level video segmentation and on the use of *HMMs*. The final skim is a sequence of shots which are obtained as observations of the *HMMs* corresponding to story units, and by a set of salient features which roughly estimates the informativeness of shots, depending on film genres. The effectiveness of the proposed solution has been compared and demonstrated in terms of informativeness and enjoyability on a large movie set coming from different genres. From the user study we can conclude that skims generated using the proposed method are not as good as manually skims, but have considerably higher quality than skims generated using a random sampling method.

Ongoing work aims at broadening the set of available salient features for different video genres, for example modifying the already described salient feature related to human faces according to the percentage of music/silence/speech inside each shot. The same salient feature based on audio classification could be useful to skim music programmes, for example to isolate songs from the other material in the show. Further applications of the proposed method to video mash-up are also envisaged and currently under investigation.

Acknowledgments

The authors would like to thank the reviewers for the accurate comments and PhD. student Luca Canini for priceless help during the experimental evaluation.

References

[1] Y.-F. Ma, L. Lu, H.-J. Zhang, and M. Li, "A user attention model for video summarization," in *Proceedings of the 10th ACM International Multimedia Conference and Exhibition*, pp. 533–542, Juan Les Pins, France, December 2002.

[2] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, "Video summarization and scene detection by graph modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, pp. 296–304, 2005.

[3] M. A. Smith and T. Kanade, "Video skimming and characterization through the combination of image and language understanding techniques," in *Proceedings of the IEEE International Workshop on Content-Based Access Image Video Data Base*, pp. 61–67, January 1998.

[4] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[5] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, Cambridge, UK, 1999.

[6] Y. Wang, Z. Liu, and J. C. Huang, "Multimedia content analysis," *IEEE Signal Processing Magazine*, vol. 17, no. 6, pp. 12–36, 2000.

[7] L. Xie, S.-F. Chang, A. Divakaran, and H. Sun, "Structure analysis of soccer video with hidden Markov models," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '02)*, vol. 4, pp. 4096–4099, Orlando, Fla, USA, May 2002.

[8] B. T. Truong and S. Venkatesh, "Video abstraction: a systematic review and classification," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 3, no. 1, p. 3, 2007.

[9] S. Benini, P. Migliorati, and R. Leonardi, "Hierarchical structuring of video previews by leading-cluster-analysis," *Signal, Image and Video Processing*, 2010.

[10] Y. Gao, W.-B. Wang, J.-H. Yong, and H.-J. Gu, "Dynamic video summarization using two-level redundancy detection," *Multimedia Tools and Applications*, vol. 42, no. 2, pp. 233–250, 2009.

[11] N. Omoigui, L. He, A. Gupta, J. Grudin, and Sanocki, "Time-compression: systems concerns, usage, and benefits," in *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 136–143, May 1999.

[12] A. Amir, D. Ponceleon, B. Blanchard, D. Petkovic, S. Srinivasan, and G. Cohen, "Using audio time scale modification for video browsing," in *Proceedings of the 33rd Hawaii International Conference on System Sciences*, vol. 3, pp. 3046–3055, January 2000.

[13] Z. Li, G. M. Schuster, and A. K. Katsaggelos, "Minmax optimal video summarization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 10, pp. 1245–1256, 2005.

[14] Y.-F. Ma and H.-J. Zhang, "A model of motion attention for video skimming," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '02)*, vol. 1, pp. 129–132, Rochester, NY, USA, September 2002.

[15] G. Evangelopoulos, A. Zlatintsi, G. Skoumas, et al., "Video event detection and summarization using audio, visual and text saliency," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '09)*, pp. 3553–3556, Taipei, Taiwan, April 2009.

[16] J. Nam and A. T. Tewfik, "Video abstract of video," in *Proceedings of IEEE 3rd Workshop on Multimedia Signal Processing*, pp. 117–122, September 1999.

[17] A. Hanjalic and H. Zhang, "An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 8, pp. 1280–1289, 1999.

[18] X. Zhu, X. Wu, J. Fan, A. K. Elmagarmid, and W. G. Aref, "Exploring video content structure for hierarchical summarization," *Multimedia Systems*, vol. 10, no. 2, pp. 98–115, 2004.

[19] Y. H. Gong and X. Liu, "Video summarization using singular value decomposition," in *Proceedings of the of International Conference on Computer Vision and Pattern Recognition (CVPR '00)*, vol. 2, pp. 174–180, 2000.

[20] "Rushes FP6-045189," http://www.rushes-project.eu/.

[21] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and TRECVid," in *Proceedings of the 8th ACM International Multimedia Conference and Exhibition (MIR '06)*, pp. 321–330, New York, NY, USA, 2006.

[22] E. Rossi, S. Benini, R. Leonardi, B. Mansencal, and J. Benois-Pineau, "Clustering of scene repeats for essential rushes preview," in *Proceedings of the 10th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS '09)*, pp. 234–237, London, UK, May 2009.

[23] Q. Huang, Z. Liu, A. Rosenberg, D. Gibbon, and B. Shahraray, "Automated generation of news content hierarchy by integrating audio, video, and text information," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '99)*, vol. 6, pp. 3025–3028, March 1999.

[24] W.-T. Peng, Y.-H. Chiang, W.-T. Chu, et al., "Aesthetics-based automatic home video skimming system," in *Advances in Multimedia Modeling*, vol. 4903 of *Lecture Notes in Computer Science*, pp. 186–197, 2008.

[25] Y. Takahashi, N. Nitta, and N. Babaguchi, "Video summarization for large sports video archives," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '05)*, pp. 1170–1173, Amsterdam, The Netherlands, July 2005.

[26] H. Sundaram, L. Xie, and S.-F. Chang, "A utility framework for the automatic generation of audio-visual skims," in *Proceedings of the 10th ACM International Multimedia Conference and Exhibition*, pp. 189–198, Juan Les Pins, France, 2002.

[27] T. Tsoneva, M. Barbieri, and H. Weda, "Automated summarisation of narrative video on a semantic level," in *Proceedings of the IEEE International Conference on Semantic Computing (ICSC '07)*, Irvine, Calif, USA, September 2007.

[28] N. Dimitrova, M. Barbieri, and L. Agnihotri, "Movie-in-a-minute," in *Proceedings of the 5th IEEE Pacific-Rim Conference on Multimedia (PCM '04)*, Tokyo, Japan, December 2004.

[29] A. Hanjalic, R. L. Lagendijk, and J. Biemond, "Automated high-level movie segmentation for advanced video-retrieval systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 580–588, 1999.

[30] M. M. Yeung and B.-L. Yeo, "Time-constrained clustering for segmentation of video into story units," in *Proceedings of the 13th International Conference on Pattern Recognition (ICPR '96)*, vol. 3, pp. 375–380, Vienna, Austria, August 1996.

[31] S. Benini, A. Bianchetti, R. Leonardi, and P. Migliorati, "Video shot clustering and summarization through dendrograms," in *Proceedings of the Image Analysis for Multimedia Interactive Services (WIAMIS '06)*, pp. 19–21, Incheon, South Korea, April 2006.

[32] S. Jeannin and A. Divarakan, "MPEG7 visual motion descriptors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 720–724, 2001.

[33] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 1, pp. 511–518, 2001.

[34] N. Adami and R. Leonardi, "Identification of editing effect in image sequences by statistical modelling," in *Proceedings of the Picture Coding Symposium (PCS '99)*, pp. 157–160, Portland, Ore, USA, April 1999.

[35] "Internet movie database," http://www.imdb.com/.