Tackling Action-Based Video Abstraction of Animated Movies for Video Browsing

Bogdan IONESCU[a], Laurent OTT[cb], Patrick LAMBERT[b]
Didier COQUIN[b], Alexandra PACUREANU[a], Vasile BUZULOIU[a]
[a] University "Politehnica" of Bucharest, LAPI, 061071, Bucharest, Romania
BIonescu, APacureanu, Buzuloiu@alpha.imag.pub.ro
[b] University of Savoie, LISTIC, BP.806, 74016 Annecy-Cedex, France
Patrick.Lambert, Didier.Coquin@univ-savoie.fr
[c] University of Strasbourg, LSiiT, BP.10413, 67412 Illkirch-Cedex, France
Ott@lsiit.u-strasbg.fr

May 11, 2009

Submitted to SPIE International Journal on Electronic Imaging

ABSTRACT

In this paper we address the issue of producing automatic video abstracts in the video indexing context of animated movies. For a quick browse of the movie's visual content, we create first a storyboard-like summary, which follows the movie events by providing one key frame for each particular scene. To capture the shot visual activity, we use histograms of cumulative inter-frame distances and key frames are adapted to histogram mode distribution. Secondly, for a "sneak peak" of the movie's exciting action content, we propose a trailer-like video skim, which aims at providing only the most interesting parts of the movie. Our method is based on a relatively standard approach, thus highlighting action through the analysis of the movie rhythm and visual activity information. To fit every movie content, including predominant static movies or movies without exciting parts, the concept of action gets tuned according to movie average rhythm. The efficiency of our approach

has been confirmed through several user studies, directly by the "consumers of the product", i.e. the end-users.

Keywords: video highlight, video skim, video summary, video trailer, action content, visual activity, user evaluation campaign, animated movies.

1 Introduction

Thanks to recent advances in multimedia technology, e.g. higher storage capacities, faster networking, better performance portable devices, video information becomes accessible to almost everyone. Video information is massive data, for instance, only one minute of video is the equivalent of up to 1.800 static images. Nowadays, video databases (e.g. news videos, sport videos, music videos, etc.) may contain as many as thousands and thousands of videos.

This information is not a problem for the available computational infrastructure, which can handle such huge amounts of data successfully, but for the human factor. For instance, browsing the database in the search for a specific movie or a particular scene, can be a tricky task, as it requires to visualize movie contents. Visualizing each movie, is, first of all, inefficient, due to data redundancy, and also, can be very time consuming, as one may require months to proceed all the video footage.

For addressing this issue, *movie abstracts* are one efficient solution. A movie abstract is a compact representation of the original video, significantly shorter, which ideally preserves at least some of the essential parts of the original video [1]. Therefore, a movie abstract provides a fast and efficient representation of the original video content.

There are two fundamentally different kinds of video abstracts. The still-image abstracts, known as *video summaries*, are a small collection of salient images (i.e. key frames) that best represent all the underlying content or only certain scenes of interest. Basically, the existing methods differ in the way the key frames are extracted [6] [22]. For instance, in [7] key frames are extracted using

derivatives on a curve of characteristic frame vectors, in [8] the key frame extraction process is compared to TF-IDF (Term-Frequency Inverse Document Frequency) text searching mechanism, and key frames are selected at shot level as frames having a reduced frame coverage of the rest of the movie, [9] uses the Sequence Reconstruction Error (SRE) which measures the capacity of the summary of reconstructing the original movie contents, while [11] tackles the key frame extraction process with the help of mathematical modeling.

On the other hand, the moving-image abstracts, or *video skims*, consist on a collection of image sequences. Therefore, a video skim is, itself, a video sequence. One simple and straightforward method to create a movie skim is to include some of the neighborhood frames (i.e. a frame interval) of the key frames extracted using a video summary technique [13]. However, the complexity of a video skim yields some more developed approaches. The existing video skimming techniques are addressing two different approaches, thus: *summary sequences*, which are classic abstracts covering the entire movie contents, and *movie highlights*, which only summarize some of the most interesting parts of the movie [14]. Video highlighting techniques are related to the characteristics of the events, which are to be considered as representatives for the underlying movie content [6]. Therefore, the existing approaches are application dependent.

Several approaches have been tackled, for instance [15] addresses the skimming of sport videos and the video highlight extraction is performed on the events which trigger particular reactions from the public, e.g. applause or cheering, [16] considers only the movie segments which produce certain reactions from the narrator/commentator, e.g. exciting, [17] uses for the video highlight, only the movie segments which are emphasized by specific editing techniques, e.g. a high cut frequency, presence of text or the replay of certain scenes, or [18] where the selected movie segments are the ones which have specific viewing patterns, e.g. most often viewed by the user.

Having the two available abstraction techniques, i.e. video summaries and video skims, due to their major differences in terms of the information they provide, one may be tempted to choose one as being the most reliable and complete, i.e. video skims. In fact, in the context of indexing,

both types of abstracts are as much necessary, being efficient in almost complementary situations. Both video abstraction techniques have been used extensively with content-based video indexing systems [2], for now more than one decade, to reduce the browsing time, to improve the quality of the search as well as to reduce the computational complexity by replacing the original movie in the processing steps.

Static abstracts, or video summaries, are easy to compute, as they contain only the visual information of the movie. Depending on the application, the computational complexity can be very reduced without significant quality loss. For instance, a quick summary can be produced by retaining one image per shot (e.g. central image, random image, median image, etc.). Also, video summaries are easy to visualize, being a collection of static images. In this case, there is no need for synchronizing or temporizing the data. Finally, video summaries can be close to the storyboard of the movie (images showing the order of the scenes of the film), thus providing information about the movie key scenes [34]. Therefore, certain types of movie summaries may come in handy when the user wants to make a fast impression, "in the blink of an eye", on the movie contents, and does not want to spend precious time visualizing some movie clips.

On the other hand, the possibly higher computational effort during the skimming process, pays off during the playback time. It is obvious that it is more natural and more interesting for the user to watch a trailer than a slide show of static images [1]. Video skims make more sense, as they provide one movie with fundamental information, which is the dynamic/motion content. Video skims, and particularly video highlights, are efficient when the user wants to quickly browse for the movie action content, as they tend to remove unexciting and redundant movie parts [19]. Their main disadvantage is, in general, their higher computational complexity, compared to video summaries, and the need of tackling the issue of synchronizing sound and image.

What comes next in this paper is organized as follows. Section 2 reviews the literature on the subject of the paper and connects it to our contributions. Section 3 aims at familiarizing the reader with the characteristics of the animation domain. In Section 4 we present an overview of our

approach. Section 5 covers the video temporal segmentation. In Section 6 we describe the proposed approach for highlighting action content using movie rhythm, while in Section 7 we discuss the use of visual activity information to enhance the previous determined action information. Section 8 deals with the construction of the video trailer and the storyboard-like summary. In Section 9 we present and discuss the experimental results. Finally, the conclusions and future work are discussed in Section 10.

2 Related work

Movie trailers are a particular case of video highlights. They are formally defined as being a short promotional film composed of clips showing highlights of a movie due for release in the near future. In general, they present some of the most exciting action parts of the movie, in order to capture the viewer attention. Currently, the manual production of this type of video abstract is quite a costly creative process, in terms of human resources and also, as it is time consuming. For instance, to produce a trailer for a short animated movie (less than 10 minutes), an expert may need as long as up to 5-6 hours [3]. Therefore, automatic approaches are more than welcome.

Unfortunately, building automatic movie trailers, similar to the ones, a human may generate, is still an open issue. This requires the fully understanding of the movie grammar, which is limited with today's scientific progress [24]. However, to cope with this issue, the few existing approaches are either developed in restraint conditions, e.g. they are application dependent and use "a priori" data or the expertise of the domain (close captions, electronic guides, etc.), or aim to only assist the manual movie trailer production.

Little work has been reported to date on automatically generating movie trailers [23] [25] [24] [26]. The method proposed in [23] first selects clips from the original video based on the detection of special events: dialogs, shots, explosions, text occurrences and general action indicators. An editing model is used then to assemble those clips into the movie trailer, e.g. dialog clips are merged with other dialog clips using gradual transition, while dialog clips and event clips are merged together

using cuts.

The approach proposed in [25] is based on sets of rules or grammars which encapsulate the theory of the film composition. They analyze action movies in terms of shot change detection, the MPEG-7 measurement of motion activity and a set of audio features based on the energy of the audio signal, which combined, form a feature, referred to as movie tempo. The movie trailer is composed of those video shots which have a tempo value greater than a certain threshold.

A similar approach is proposed in [24], in which shots are selected from the movie to assist in the creation of video trailers. A set of audiovisual features are extracted from the movie temporal structure (video transitions), audio track (type) and motion information (motion intensity, camera movement), to model the characteristics of typical trailer shots. The relevant shots are then obtained through a Support Vector Machine classification process. One particularity of this approach consists in its validation, which is performed by comparing the proposed trailers against genuine commercial trailers.

Another example is the method proposed in [26], which tackles the production of TV program trailers (short video clips to advertise the program) using descriptions from electronic program guides. Two methods are discussed. The first one, is based on the sentence similarity between the close captions and the introductory text of the target program. The similarity is evaluated with Bayesian belief networks. The second method extracts several sentences which have the same textual features as those of a general introductory text, and determines the corresponding video sections.

In this paper we propose an unified approach for generating *automatic movie trailers* and *storyboard-like summaries* in the context of content-based retrieval of animated movies. This is an extension of the work proposed in [27] [31], and pushes forward the efforts of developing tools for accessing the contents of animated movies at a semantic level [21] [10], by tackling the content-based browsing issue.

The "International Animated Film Festival", managed by CITIA [3], is an yearly festival, which has taken place in Annecy (France), since 1960. It is one of the major events in the worldwide animated movie entertainment, being the equivalent of the "International Festival of Cannes" in the animation industry. Every year, hundreds of movies, from all over the world, are competing. A few years ago, CITIA had the initiative of digitizing most movies, to compose one of the first digital animated movie libraries. Today, this library accounts for more than 31.000 movie titles, 22.924 companies and 60.879 professionals, which are to be available, online, for a general and professional use, through the "Animaquid" [20] indexing system. For the moment, the existing indexing tools, are limited to use only the textual information mainly provided by movie authors, which in many cases does not totally apply to the rich artistic content of the animated movies from [3]. The artistic content is strongly related to visual information, which gets poorly described with textual information.

Therefore, having for each movie, a trailer-like abstract or a collection of key frames which follow the movie event evolution available, should be an ideal way of accessing its content in a content-based retrieval system, like "Animaquid" [20].

Our movie trailer approach is based on highlighting the movie action segments by analyzing the movie, both at inter-shot and inter-frame level. At inter-shot level action is detected by selecting movie segments with a high frequency of video transitions over a certain time-window. A more accurate detection is performed at frame-level, at which the visual activity is captured with histograms of cumulative inter-frame distances. For the proposed video summary, the key frames are extracted according to each scene, in the storyboard manner. On the other hand, the movie trailer is produced for each action segment by retaining, a percentage of its frames, which gets tuned according to each shot visual activity level.

Globally, this is not a particularly new idea, being more or less adopted by some of the previous mentioned approaches, e.g. [25]. The novelty of this work is rather in the efficiency of this relatively standard approach, when transposed and adapted to the specificity of the animation domain. To

solve the problem of producing trailers for movies with a predominant static content, for which the notion of trailer is rather ambiguous, we adapt the action detection to movie contents, as we cannot speak of action segments or exciting scenes. The performance of our approach has been confirmed through several user studies, directly by the "consumers of the product", i.e. the end-users.

3 Peculiarity of animated movies

Animated movies from CITIA [3] are different from classic animation movies (i.e. cartoons) or from natural movies, in many respects [10]. We present some of the main issues:

- *length*: the target animated movies are short-reel movies, e.g. less than 17 minutes;
- *events*: we mainly deal with fiction or abstract movies, therefore there are no physical rules;

Figure 1: Various animation techniques (from left to right and top to bottom): paper drawing, object animation, 3D synthesis, glass painting, plasticine modeling and color salts CITIA [3].

- *characters*: if any, can take any shape or color;
- *motion*: could be discontinuous, depending on the animation technique, while the predominant motion tends to be the object/character motion [33];
- *animation techniques*: a large variety of animation techniques are employed, see Figure 1;
- *visual effects*: usually, there are a lot of color effects [28];
- *artistic concepts*: the movies are artistic creations, therefore artistic concepts are used, e.g. painting concepts, theatrical concepts, etc.;

- *colors*: are selected and mixed by the artists to express particular feelings or to induce particular impressions, therefore each movie has a specific color palette [10];

- *content*: is very varied, some animation experts say that more than 30% of the animated movies from CITIA [3], cannot be summarized as their content is singular.

Some examples are illustrated in Figure 1. Despite the complexity of the content, in our approach we consider some simplifications. For instance, we deal with short animated movies, i.e. less than 15-17 minutes, which reduces the dispersion of the action segments, within the movie. Also, most of the movies are without dialog or commentary (e.g. from the 52 test movies from [10], only 8 had dialogs), therefore, in our approach, the sound is disregarded, and consequently the issue of image-sound synchronization.

4 Overview of the proposed approach

The proposed unified approach for video abstraction is based on highlighting the movie's action segments, both at shot and frame level [27] [31]. It uses several analysis steps, which are described in Figure 2.

First, the movie is divided into shots by detecting the video transitions, i.e. cuts, fades, dissolves and specific color effects. Through test shows that, in general, action content is related to video parts presenting a high frequency of shot changes over a certain time unit, that is, a high rhythm of the succeeding events. To detect it, we perform an inter-shot analysis, which consists in a four step detection algorithm (thresholding action, merging, clearing and removing segments). In our scenario, we consider action, any rhythm, which is different from the average movie tempo. This provides a rough localization, within the movie, of almost all the representative/action parts.

Another processing step, is the inter-frame analysis. This aims at providing more details on the visual activity at shot level, information which is absent with inter-shot analysis. For that, we compute cumulative inter-frame distance histograms, which capture the variability of the shot

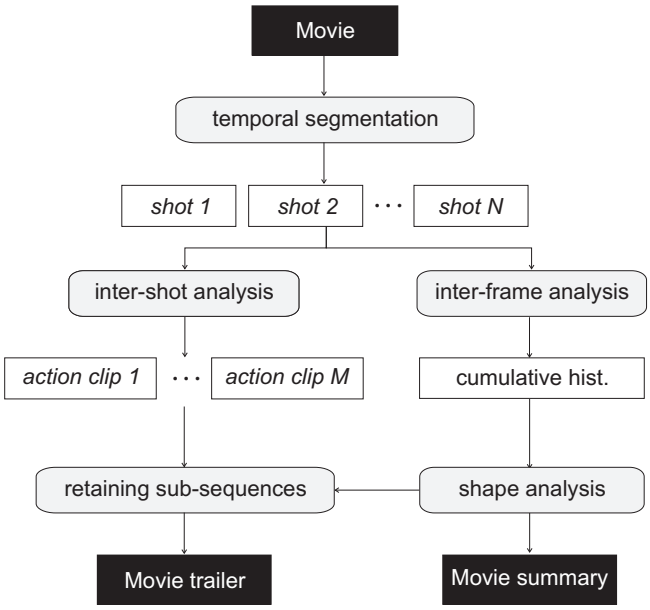

Figure 2: The diagram of the proposed video abstraction approach.

contents.

For the video summary, key frames are extracted, from each shot, using different approaches, according to the shape of the cumulative histograms. This adaptive strategy produces one key frame in general for each different visual scene, in the storyboard manner.

For the movie trailer we use, at the same time, action clips and cumulative histograms. The movie trailer is produced by retaining for each action clip, a percentage of its frames, which is tuned according to the shape of the histogram. Therefore, high activity shots have a more important contribution to the trailer than the static ones. Each processing step is discussed in the sequel of the paper.

5 Temporal segmentation

The temporal segmentation of the movie is a fundamental processing step, required by most of the existing video analysis techniques [32], as it provides the movie basic temporal unit structure, or shot structure. Roughly speaking, the temporal segmentation means detecting the video transitions, which make the connections between different shots.

The methods we use, were specially designed to cope with the peculiarity of animated movies (see Section 3). First, we detect, jointly, cuts (or sharp transitions, which are the most frequent transitions), and an animated movie specific color effect, namely SCC ("short-in-time dramatic color change", e.g. explosion, lightning, etc.). We use the histogram-based approach proposed in [28]. From the existing gradual transitions, we detect the most representatives for the animated movies, namely: fades (i.e. fade-in and fade-out) and dissolves. Fade detection is performed with an adaptation of the pixel-level statistical approach proposed in [29], while dissolve detection is based on the evaluation of fading pixels proposed in [30]. Detailed information about the detection of video transitions, as well as several experimental results are presented in [5].

Video transitions are then synchronized with the respect to the movie time axis. Video shots are determined as being the video segments which lie between two successive video transitions, according to some constraints (e.g. black frames between transitions are removed, transitions frames are removed, etc.).

To assist in the detection of the movie action clips, we construct, what we call, a visual annotation graph of the movie temporal structure. The annotation graph describes the movie temporal evolution as a time-continuous signal, of a certain arbitrary amplitude (e.g. 1), which is interrupted by transitions back and forth to zero, corresponding to the occurrence of video transitions (a particular case are the SCC color effects which are depicted as small peaks, see Figure 5). This graph provides the density of video transitions during the movie. The next section will describe the use of the annotation graph to determine action clips.

6 Highlighting action

The inter-shot analysis aims at highlighting the movie action parts, which, in general, in the animated movies from CITIA [3], are related to a high frequency of shot changes. This is a relatively standard confirmed approach which is used with the production of movie trailers [25].

Several experiments were conducted to study the correlation between, action and shot change density, which is measured with two action indicators, as follows.

6.1 Measuring rhythm

First, we define a basic indicator, related to the movie temporal structure, which measures, what we call, the *movie change rhythm*. This indicator, denoted $\zeta_T(i)$, represents the relative number of shot changes occurring within the time interval of T seconds, starting from frame at time index i. Regarding $\zeta_T(i)$ from the point of view of a discrete random variable, its distribution over the entire movie, can be estimated by computing $\zeta_T(i)$ for all the movie time windows of size T. In order to do so, we use a certain processing step τ, as described with Figure 3.

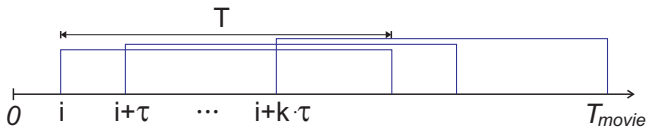

Figure 3: Measuring $\zeta_T(i)$ with a step τ (the oX axis is the temporal axis, T is the window size, k is a positive integer and T_{movie} is the movie length).

Having the distribution of $\zeta_T(i)$, we define a second action indicator further denoted *mean shot change speed*, $\bar{v}_T$, which is given by the following equation:

$$\bar{v}_T = E\{\zeta_T(i)\} = \sum_{t=1}^{T\cdot 25} t \cdot f_{\zeta_T(i)}(t) \tag{1}$$

in which $T \cdot 25$ represents the number of frames of the time window (at 25 fps) and $f_{\zeta_T(i)}$ is the probability density of $\zeta_T(i)$ given by:

$$f_{\zeta_T(i)}(t) = \frac{1}{N_T} \sum_{i \in W_T} \delta(\zeta_T(i) - t) \tag{2}$$

in which N_T is the total number of time windows of size T seconds, i is the starting frame of the current analyzed time window (containing $\zeta_T(i)$ shot changes), W_T is the set of all analyzed time windows and $\delta(t) = 1$ if $t = 0$ and 0 otherwise.

We can note that:

$$N_T = (T_{movie} - T)/\tau + 1 \tag{3}$$

in which T_{movie} is the movie length in seconds.

Defined in this way, $\bar{v}_T$ represents the average number of shot changes over the time interval T for the entire movie, being a measure of the movie global tempo [21]. High values of $\bar{v}_T$ indicate a movie with a general high change ratio, while small values typically correspond to movies with predominant long and static shots (a reduced number of scenes).

Concerning the tuning of the two action indicators, i.e. $\zeta_T(i)$ and $\bar{v}_T$, the choice of parameter T, empirically set to $T = 5s$, is discussed with subsection 6.3, as it is related to the granularity of the resulted action clips.

For determining the right value of the second parameter, thus the processing step τ, we compute $\bar{v}_T$ for a small set of representative animated movies with τ ranging from 1/25 (temporal windows are superimposing, the delay is of only one frame) to 5 ($\tau = T$, temporal windows are not superposing each other, being disjunctive), with a step of one frame, i.e. 1/25 seconds. Some of the results are depicted in Figure 4.

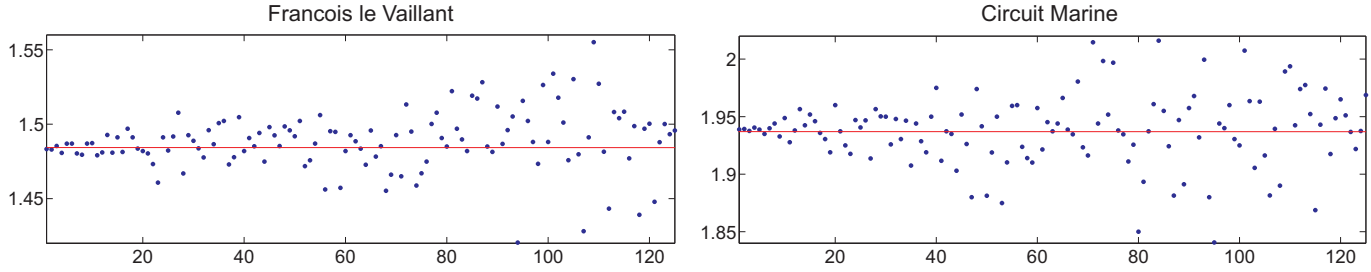

Figure 4: Evaluating $\bar{v}_T$ for different values of the processing step τ (the oY axis corresponds to $\bar{v}_T$ while the oX axis to τ, the graph average value is depicted with the red line, movies from [4]).

One may observe that, globally, $\bar{v}_T$ tends to scatter from the average value, proportionally with the increase of the step τ, and thus with the decrease of the number of particular realizations of the random variable $\zeta_T(i)$. However, for almost any interval of τ values, at any scale, some of the $\bar{v}_T$ values always stay close to the average. Therefore, we conclude that the most accurate estimation of $\bar{v}_T$ is near the average, being archived for the maximum number of time windows, and thus the smallest processing step, i.e. $\tau = 1/25$ seconds (a step of one frame).

6.2 Shot change vs. action

Having a measure of the movie shot change ratio over the time unit, $\zeta_T(i)$, as well as of the movie global change rhythm, $\bar{v}_T$, we should address the issue of how these parameters are related to action content.

For that, we have conducted an experimental test on a small set of animated movies (8 movies from Folimage [4] and Pixar Animation Company). Several people were asked to manually browse the movie contents and to classify them, if possible, into three action categories, namely: "*hot action*", "*regular action*" and "*low action*". These categories have the following meaning: "*hot action*" corresponds to movie segments with an intense action content (e.g. fast changes, fast motion, visual effects, etc.), "*regular action*" stands for common action segments (e.g. dialogs scenes, regular events, etc.) and "*low action*" is mainly for static scenes.

For each manually labeled action segment, we compute the mean shot change ratio, $\bar{v}_T$, to capture the corresponding changing rhythm. Some of the results are presented in Table 1. Then, we compute the overall $\bar{v}_T$ mean values over all the segments within each action category, as well as the standard deviation. Having these pieces of information, we determine the intervals of $\zeta_T(i)$ values which correspond to each type of action content, as $[E\{\bar{v}_T\} - \sigma_{\bar{v}_T}; E\{\bar{v}_T\} + \sigma_{\bar{v}_T}]$. The results are synthesized with Table 2.

Therefore, a movie content rich in action is represented with values of $\zeta_T(i)$ starting, around 2.8, a regular action content is represented with $\zeta_T(i)$ between 1.6 and 2.3 and low action content has values of $\zeta_T(i)$ less than 0.7. Values of $\zeta_T(i)$ which do not fall in the previously determined intervals, correspond more or less to ambiguous action contents, which are difficult to be classified in one of the three "a priori" classes.

The proposed movie trailer aims at presenting, first of all, the action segments, denoted as "hot action", but as well as some of the "regular action" parts. The direct use of the previously computed intervals of $\zeta_T(i)$, to determine the action content, could be inefficient, especially for the animated

movies which do not include "hot action" segments or even "regular action", or for some reasons, do not fall precisely in these intervals. In our approach, we prefer to sacrifice the universality of the method for the sake of an adaptive approach, which will fit every movie content. The method is described with the following.

6.3 Determining action clips

We consider, generically, an action clip, a movie segment for which the frequency of changes, i.e. the value of $\zeta_T(i)$, is greater than the movie average value, i.e. $\bar{v}_T$. In other words, we consider action everything that is different from the common movie change rhythm.

For a regular movie, i.e. a movie following the classical narrative scheme, $\bar{v}_T$ will be placed somewhere in the "regular action" interval (e.g. "François le Vaillant", "Circuit Marine", see Figure 4 and Table 2), which corresponds to action groundtruth and thus to our goal. On the other hand, for non-regular movies, e.g. predominant static movies, the notion of trailer is ambiguous, as we cannot speak of action segments or exciting scenes. However, in this case a trailer will present, not the explosive parts which are missing, but some of the movie representative parts. The proposed approach does this by adapting to the contents.

The previously determined action levels get adapted, being translated according to $\bar{v}_T$ value, thus "low action" is a rhythm below the average, "regular action" is around the average and consequently "hot action" is above the average. The main advantage of this approach is that every movie, regardless the narrative content, gets represented in the trailer with its most "uncommon" parts.

The action clips are highlighted on the visual annotation graph (see Section 5) using the following four-step algorithm:

- **a. thresholding**: first, we define a binary signal, as a function of frame index, thus:

$$action(i) = \begin{cases} 1 & \text{if } \zeta_T(i) > \bar{v}_T \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

in which i represents the index of current frame. Defined in this way, $action(i)$ captures only the movie segments which present a relatively high number of shot changes (greater than the movie average, as stated before). Using, $action(i)$, action clips are determined, on a first iteration, as time continuous intervals of 1 values. The process is illustrated in Figure 5, graph a.

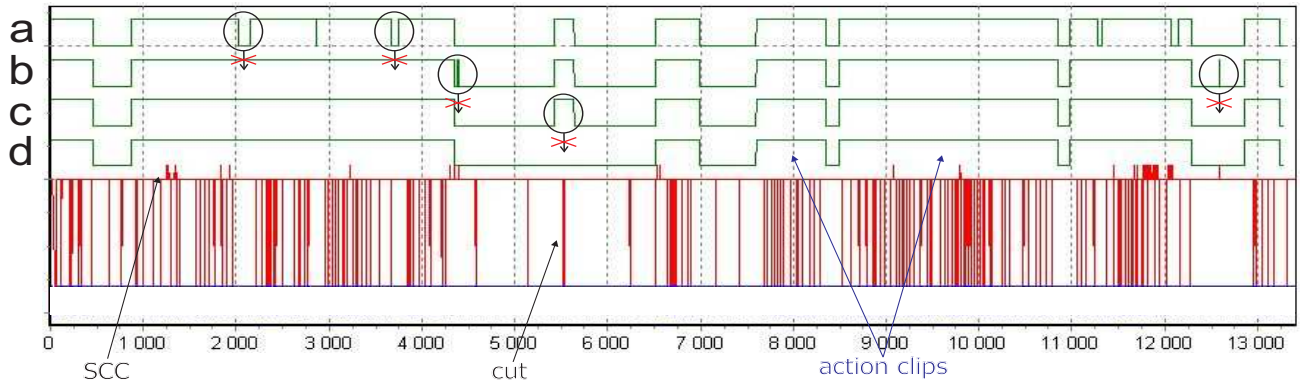

Figure 5: Action highlighting using the visual annotation graph: the oX axis is the time axis, $a - d$ denote the processing steps, the red line is the temporal annotation graph (vertical lines correspond to video transitions) and the green line represents the $action(i)$ binary signal ($T = 5$ seconds, movie "François le Vaillant" [4]).

- **b. merging**: the SCC color effects [28] are marked as action segments, as they contain exciting visual information. Then, the neighbor action clips, at a time distance below than T seconds (the size of the time window) are merged together to form a single action clip. This step allows us to remove the gaps may appear after the thresholding step and thus to reduce the over-segmentation. See Figure 5, graph b.

- **c. clearing**: the small action clips, with a length below than the analysis window, are being erased. This step allows us to remove the small spikes from the function $action(i)$ and thus to remove unnoticeable and irrelevant action segments. See Figure 5, graph c.

- **d. removing**: finally, all the action clips containing fewer than N_s video shots, with $N_s <= 4$ (empirically determined), are being removed. Those segments are very likely to be the result of false detections, containing one or several gradual transitions (e.g. a "fade-out" - "fade-in" sequence). An example is presented in Figure 5, graph d.

The results of the proposed algorithm depend on the choice of parameter T, thus the size of the temporal window. Several tests were performed on various animated movies for different values of T, namely $T \in \{1, ..., 10\}$ seconds. Globally, the value of T is related to the granularity of the action clips. Using small values of T, will result in a high density of small length action clips (action clips are over-segmented). On the other hand, a high value of T will result in fewer, but longer, action clips (action clips are sub-segmented). A good compromise between the length of the action clips and the action clip density proved $T = 5$ seconds.

Once the global action information was determined (i.e. action clips), we proceed further to refine it through the analysis of the shots visual activity, which is described in the following paragraph.

7 Analysis of shot visual activity

The aim of this processing step is to determine visual activity at shot level, as the highly variable content is, in general, exciting for the viewer. This information is used, first, to assist in the production of the video trailer by emphasizing, from the action clips, the shots with an exciting visual content (varied content), and secondly, to retrieve key frames in a storyboard-like manner, i.e. one image for each different scene/event [31].

The proposed method was inspired from color median filtering techniques, in which the output of the filter is the most representative value, thus the one which minimizes the cumulative sum of distances to all the other values. To capture the pattern of visual changes, we use histograms of cumulative inter-frame distances, as follows.

7.1 Preprocessing

First, we use some preprocessing steps. The shot is temporally sub-sampled, being processed with a step of two images, i.e. only one frame from two is retained. Then, we use a spatial sub-sampling. Each retained frame is divided into non-overlapping blocks of $n \times n$ pixels, with $n \in \{2, 3, 4, ...\}$,

and for each block only one pixel is retained. The n value is determined on the original image size to achieve a final image resolution around 100×100 pixels. As color histograms are statistical measures, spatial sub-sampling will not affect significantly the accuracy of the method. These processing steps aim at reducing the computational time.

To compute color histograms, one has to reduce the number of colors, as frames come with true color palette (16 million colors). We use a very fast uniform quantization of the RGB color space into only $5 \times 5 \times 5$ colors (5 intervals for each color axis). The typical visual quality loss which occurs with such an approach is reduced with animated movies due to their restraint variability of color hues (see Section 3).

7.2 Histogram of inter-frame cumulative distances

For each retained frame of index i, from the current shot k, we compute its color histogram, denoted $H^i_{shot_k}(c)$, in which c is the color index, $c \in \{1, ..., 125\}$. To evaluate the distance between frames, we use the classical Manhattan distance, denoted $d_M()$, which provides a good compromise between the computational complexity and the quality of the results. We use a version of this distance which is normalized to 1, thus:

$$d_M(H^i_{shot_k}, H^j_{shot_k}) = \frac{\sum_{c=1}^{125} |H^i_{shot_k}(c) - H^j_{shot_k}(c)|}{2 \cdot N_p} \tag{5}$$

in which N_p represents the number of pixels and i and j are two frame indexes.

The inter-frame cumulative distance for the current frame i of the shot k, denoted $d_{shot_k}(i)$, is given by the following equation:

$$d_{shot_k}(i) = \sum_{j \in S, i \neq j} d_M(H^i_{shot_k}, H^j_{shot_k}) \tag{6}$$

in which S is the set of the retained frames for shot k.

This measure gives us information on the correlation between the frame i and the other frames. If the cumulative distance is low, we may conclude that frame i is similar to most of the shot's frames, while if the distance is high, then the image must be different from most of the frames.

To be able to compare the histogram of cumulative distances for different shots, we use a normalized version of the distance $d_{shot_k}(i)$, which corresponds to the average distance, thus:

$$D_{shot_k}(i) = \frac{d_{shot_k}(i)}{Card(S) - 1} \tag{7}$$

in which $Card()$ returns the size of a set.

The histogram of cumulative inter-frame distances, $\aleph^D_{shot_k}$ is computed after quantifying $D_{shot_k}(i)$ values into N_b bins, denoted $D^q_{shot_k}(i)$, in which $i \in S$ and $q = 1, ..., N_b$. $\aleph^D_{shot_k}$ is further determined as:

$$\aleph^D_{shot_k}(d_q) = \sum_{i \in S} \delta(D^q_{shot_k}(i) - d_q) \tag{8}$$

in which S is the frame set for the shot k, d_q is a quantified value of the normalized cumulative inter-frame distance, q represents the bin index and $\delta(x) = 1$ if $x = 0$ and 0 otherwise. A good tradeoff between computational complexity and the precision of the histogram is $N_b = 100$.

Further on, we use the analysis of histogram shape to measure how visual activity is related to the distribution of cumulative distances within the shot.

7.3 Visual activity vs. histogram shape

After observing and analyzing several examples of histograms of cumulative inter-frame distances for a large variety of animated movies, we conclude that, despite the diversity of histograms, they can be projected basically into only a limited number of patterns, which are related to the type of shot content. We identify four classes, thus:

- **pattern 1** - *histograms with small distance*: all the values of the cumulative distance are small and therefore there is a reduced variability of the visual content (shot content is almost constant);

- **pattern 2** - *histograms with both small and high distances*: most of the cumulative distances are small, but there are a few frames which are very different from the others. This scenario

corresponds to shots in which the visual content is mainly constant, but with some important visual changes (e.g. a scene with a moving object/character);

- **pattern 3** - *multi-modal histograms*: the shot contains different groups of similar frames. This scenario corresponds in general to several static scenes which are connected by camera motion;

- **pattern 4** - *single-mode histograms*: the histogram has only one mode, but the cumulative distances are high. Such shots are composed of many different frames suggesting a constantly changing content which may result from continuous motion or due to the use of special color effects.

Some examples of histogram shapes and their corresponding shot contents are presented in Figure 9. For instance, a histogram of pattern 2 corresponds to a shot which contains a predominant group of similar content images, e.g. a focus on a character, as well as some important changes caused by a camera zoom-in motion (see shot $[8612 - 8657]$ from movie "Ferrailles" in Figure 9), while a histogram of pattern 3 corresponds to a shot with several groups of similar frames, which result from a 3D camera motion with several focuses and delays on some interest points of the scene (see shot $[78 - 735]$ from movie "The Buddy System" in Figure 9).

Figure 6 depicts the spatial distribution of the frames from the previous shots, according to visual similarities. We use a 2D projection of the multidimensional feature space determined by frame color histograms $H^i_{shot_k}$. The axis are determined by the first two principal components of histogram data, as they account for as much of the variability in the data as possible [35]. Each frame corresponds to a point in the transformed feature space.

Comparing Figure 9 and 6, one may observe that the visual similarity between the frames is well represented with the cumulative inter-frame histogram $\aleph^D_{shot_k}$. For instance, the shot $[8612 - 8657]$ from movie "Ferrailles" presents a group of similar frames (see the blue circle in Figure 6), which correspond to the only mode of the histogram $\aleph^D_{shot_k}$ (see Figure 9), as well as some different frames

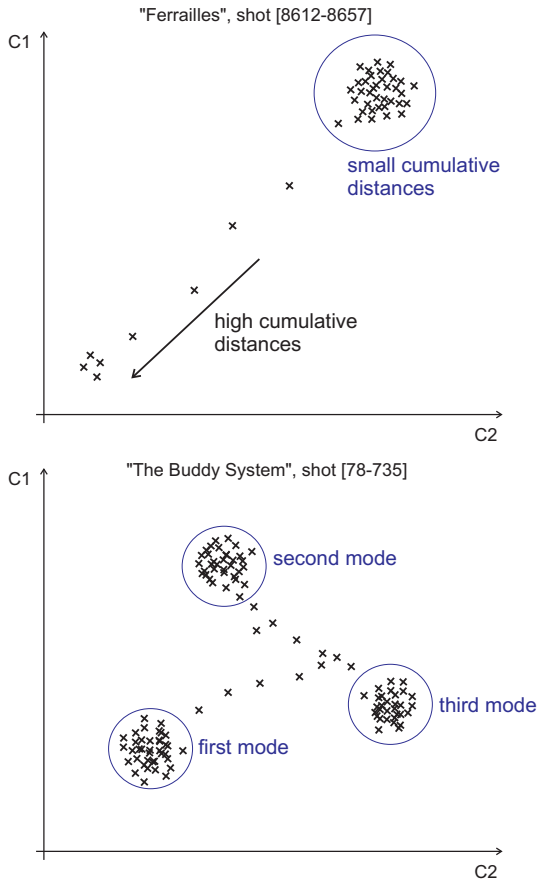

Figure 6: Spatial distribution of frames in the feature space formed by color histograms (the vertical and horizontal axis correspond to the first two principal components of histogram data, each point is a frame).

which correspond to high cumulative distances. The shot $[78 - 735]$ from movie "The Buddy System" presents three groups of similar frames (see the blue circles in Figure 6) which correspond to the three modes of the histogram $\aleph^D_{shot_k}$, and some different frames which correspond, as for the previous case, to high cumulative distances (see Figure 9).

7.4 Determining histogram shape

The previously determined patterns of the histogram $\aleph^D_{shot_k}$ are detected using multimodal histogram analysis. A histogram is considered to be a *histogram with small distances* (pattern 1), if the maximum cumulative distance over the shot frames is below than a certain threshold, thus:

$$Max_{|i \in S}\{D_{shot_k}(i)\} < \tau_1 \tag{9}$$

in which i is the frame index, S is the set of the retained frames for shot k and τ_1 is a threshold which was set empirically to 0.12.

Histograms with, both, small and high distances (pattern 2) are determined using some basic statistics of D_{shot_k} values. The idea is to position the mean value of D_{shot_k} with respect to the minimum and maximum values. We use the following test:

$$Mean\{D_{shot_k}()\} - Min\{D_{shot_k}()\} < \frac{1}{\tau_2} \cdot (Max\{D_{shot_k}()\} - Min\{D_{shot_k}()\}) \quad (10)$$

Therefore, a shot contains both small and high distance values if the previous condition is satisfied, thus if the gap between minimum and mean D_{shot_k} values is less than a fraction of the gap between maximum and minimum D_{shot_k} values. The threshold τ_2 was set to 5 after several experimental tests.

If a histogram fails to comply with one of the previous cases, we search for significant histogram peaks using an adaptation of the approach proposed in [36]. At a coarse level, we first retain all the histogram local maxima. These values are then filtered in three steps:

- the first step consists in removing all the insignificant values, i.e. with an amplitude less than 5% of the maximum peak;
- the second step removes neighbor local maxima, e.g. at a distance less than 3 bins. In this case, from two neighbor peaks, we retain only the highest peak;
- the final step, assures that the gap between two peaks is well important, which happens if the following condition is true, thus:

$$\frac{2 \cdot \bar{\aleph}}{\aleph^D_{shot_k}(p_1) + \aleph^D_{shot_k}(p_2)} <= 0.75 \quad (11)$$

in which p_1 and p_2 are the histogram bins corresponding to the two peaks and $\bar{\aleph}$ is the histogram average value given by:

$$\bar{\aleph} = \frac{1}{p_2 - p_1 + 1} \cdot \sum_{b=p_1}^{p_2} \aleph^D_{shot_k}(b) \quad (12)$$

where b is a histogram bin. If the condition is false, then from the two peaks we remove the smallest one.

The remaining peaks are, in general, related to histogram modes [36]. Therefore, if the number of peaks is greater than one, then, the histogram is a *multi-modal histogram* (pattern 3), while if only one peak is retained, then the *histogram is single-mode* (pattern 4).

8 Video abstraction

Once the action information is retrieved, we proceed with video abstraction. As stated before, we aim at providing a tool for improving and enhancing browsing video contents in the context of content-based retrieval of animated movies with the Animaquid Indexing System [20].

Due to the fact that static and dynamic abstracts are both necessary with video indexing systems, being efficient in different scenarios (see Section 1), we tackle both approaches. For a quick browse of the visual content, we constitute a storyboard-like summary, which follows the movie events by providing each particular movie scene with one key frame, while for a "sneak peak" of the movie action content we propose a trailer-like video skim, which provides only the most interesting action parts. The algorithms are presented with the following sub-sections.

8.1 Adaptive summary construction

The proposed video summary aims at presenting one image for each individual movie scene, in the storyboard manner. The key frames are extracted at shot level and the number of key frames is adapted to the variability of the shot visual content. Basically, we extract one key frame for each group of similar content images. This is done by taking into account the shape of the histogram $\aleph^D_{shot_k}$.

For a shot with a cumulative histogram of pattern 1, i.e. small distances, and thus with low visual variability, we extract only one key frame. If $\{frame_i\}$, with $i = 1, ..., N$ (N is the number

of frames) denotes the frame set of the current shot, then, the key frame, $frame_k$, is extracted according to the following equation:

$$k = argmin|_{i \in \{1,\ldots,N\}}(D_{shot}(i)) \tag{13}$$

in which $D_{shot}(i)$ is the cumulative inter-frame distance given by equation 7. Therefore, the key frame is the median image in terms of cumulative distances.

A shot with both small and high distances (pattern 2) is represented with two key frames. This type of video shot contains mainly a group of similar frames, as well as some different content images. To capture the content with a reduced visual variability, the first key frame is the median image, selected according to equation 13. The second key frame, aims at representing the changing content and is selected as $frame_l$, with l given by:

$$l = argmax|_{i \in \{1,\ldots,N\}}(D_{shot}(i)) \tag{14}$$

This frame is theoretically the most different one, providing the maximal cumulative distance over all the other frames.

For shots with cumulative histograms of pattern 3, i.e. multi-modal, which contain different groups of similar content frames, we analyze the histogram peak repartition (see Section 7.4). The idea is to extract one key frame for each individual group of similar pictures. To do so, we select one key frame for each histogram peak, as being the median image given by equation 13 when applied only to the frames which contributed to the peak value of the histogram.

Finally, single-mode histograms are represented with two key frames. In this case, the shot presents a high variability. This is the most difficult case, because a changing content requires a large number of key frames. However, selecting many images, is not always efficient, due to the probability of capturing transition images. We use a compromise and the key frames are selected with the same strategy as for the histograms of pattern 2, thus according to equations 13 and 14 (the most common image and the most different one).

The video summary preserves the temporal evolution of the movie, key frames being synchronized respecting time. Some experimental results are presented in Section 9.

8.2 Trailer production

To produce a video highlight, similar to the concept of a video trailer, we use a simple and efficient straightforward approach. It basically consists in summarizing the previously detected action clips. As stated before, with this approach, the trailer captures the movie most "uncommon" parts (see Section 6.3). The movie trailer is computed thus:

$$trailer = \bigcup_{m=1}^{M} \bigcup_{n=1}^{N_m} seq_{p\%}^{n} \tag{15}$$

in which M is the number of action clips, N_m represents the number of video shots within the action clip m, $seq_{p\%}^{n}$ is an image sequence which contains $p\%$ of the shot n frames. As the action takes place most likely in the middle of a shot, the sequence is shot centered. Retaining a number of frames according to the shot length, provides longer shots with more details, which are more valuable as they contain more information.

The choice of parameter p is related to histogram $\aleph_{shot_k}^{D}$. We adapt the amount of the retained shot information to its visual activity. Therefore, for shots with a cumulative histogram of pattern 1 or 2, which contain similar color information, we use a smaller value of p, around 15%. On the other hand, for shots with cumulative histograms of patterns 3 or 4, which contain much more action information, we use $p = 35\%$. The values of p were empirically determined after the manual analysis of several animated movies. The constraint is, first of all, to assure the visual continuity of the trailer, as well as to preserve an optimal trailer length. Several experimental results are presented in the next section.

9 Experimental results

The evaluation of video abstraction techniques is in general a subjective task, as it mainly relies on human perception. A consistent evaluation framework is actually missing, especially due to the constraints of this process, e.g. for a certain video sequence one may produce, not one, but many abstracts to cope with some quality constraints, an objective groundtruth for the evaluation is often missing, comparing different abstracts proves to be a difficult process, as even for humans it is difficult to decide whether one video abstract is better than another.

Most of the existing video abstraction methods propose their own evaluation strategy, which often lacks of consistency and universality. However, the existing approaches can be grouped in three major categories, namely: the description of the results, the use of objective metrics and user studies [6].

With the first approach, the proposed video abstraction technique is tested on a reduced set of video sequences and the results are presented and discussed to prove their efficiency. This is the most popular and simple form of evaluation, but lacks, from experimental justifications, of why the proposed technique would still work outside the few described video samples. The second approach proposes to quantify the quality of the results with some numerical scores, e.g. the sequence reconstruction error, some fidelity functions, etc. Also, in this case, there is no experimental justification for whether the metric maps well to human judgement regarding the quality of the abstract, or even so, if this guarantees the relevance of the abstract.

To test the efficiency of our approach, we jointly provide some manual descriptions of the results and a user study, which is probably the most useful and realistic form of evaluation, as it involves the human user in the evaluation process. In this way, the quality of the abstract gets judged, directly by the "consumer of the product" and the user's level of satisfaction is proportional to the quality of the results. As several different people are involved in the evaluation process, the subjectivity is reduced and thus the results are closer to reality.

User studies provide, in general, only non-visual results, therefore, first, we perform a manual analysis of the results for several representative videos while providing some visual examples. The second evaluation is an objective evaluation achieved through an evaluation campaign. We have conducted a user study involving 27 people (students, didactic personnel and several animation experts, with ages varying from 21 to 49). The tests were performed on a selection composed of 10 animated movies from CITIA [3], namely: "Casa" (6min15s), "Circuit Marine" (5min35s), "Ferrailles" (6min15s), "François le Vaillant" (8min56s), "Gazoon" (2min47s), "La Bouche Cousue" (2min48s), "La Cancion du Microsillon" (8min56s), "Le Moine et le Poisson" (6min), "Paroles en l'Air" (6min50s) and "The Buddy System" (6min19s).

The test protocol consists in showing the participants, first, the entire movie and then, the proposed movie trailer and video summary. The video summaries are presented as slideshows (1 image/1.5 seconds). After visualizing each abstract, the participants were asked to answer several questions concerning the quality of the proposed abstracts. The answers are quantified into several degrees for which a score is assigned.

For each sequence, we ,then, compute the average score, which provides an overall appreciation, as well as its standard deviation, which provides information about the non-concordance of the results. The results are described with the following sections.

9.1 Adaptive summary evaluation

Several results are illustrated in Figure 7 and 9. Overall, the proposed video summarization strategy gives a good representation of the shot contents, key frames being selected according to the complexity of each shot.

Figure 9 presents some examples of shot summaries for different patterns of cumulative histograms. For instance, multi-modal histograms conclude with one key frame for each individual group of similar pictures, e.g. the shot $[78 - 735]$ from the movie "The Buddy System", which contains a 3D continuous camera motion with several focuses on some interesting points of the

Figure 7: Comparison between the proposed adaptive summary and the abstract obtained with one image per shot (middle frame): shot boundaries are depicted with red vertical lines, the shot number is depicted with white (extract from the full summary of movie "Circuit Marine" [4], temporal order from top to bottom and left to right).

scene, is summarized with one representative frame for each focus point, or the shot $[4847 - 4906]$ in which the key frames correspond to each of the two different scenes. On the other hand, visual effects and constant changes may lead to some artificial histogram modes and thus to redundant key frames, as it is the case of frame 2561 extracted from shot $[2204 - 3172]$, movie "Le Roman de Mon Ame". Shots containing only one group of similar frames and several visual changes, are summarized with one common image and one image which captures the variability of the content, see shot $[1167 - 1409]$ from the movie "Le Roman de Mon Ame" or shot $[8612 - 8657]$ from movie "Ferrailles". Due to histogram invariance, constant shots are summarized with only one image,

despite any object motion of other small movements, see shot $[3246 - 3433]$ from movie "Gazoon".

Figure 7 compares the proposed video summary against a standard approach, i.e. extracting one image for each shot (the middle frame). The adaptive summary provides more details for a changing content (e.g. shots 1, 6, 12, 19, 24, 32 from Figure 7), while only one frame is extracted from constant shots (e.g. shots 2, 27, 31, 40, 47 from Figure 7). Overall, this results in a picture story of the entire movie action content.

However, a less subjective evaluation is given by the user campaign. The results are presented in Figure 8. For the video summary, the evaluation consisted in answering to two questions, thus:

- the first question concerns the quality of the movie content representation with the abstract, namely: "*Do you think that the proposed summary fits well the movie content ?*". For the evaluation the score range from 0 to 10, with the following signification: 0-don't know, 1,2-not at all, 3,4-very little, 5,6-partially, 7,8-almost entirely, 9,10-entirely;

- the second question concerns the length of the summary, thus: "*What do you think of the length of the video summary ?*". For this question the scores range also from 0 to 10 having the following meaning: 0-don't know, 1.2-too short, 3,4- short, 5,6-appropriate, 7,8-long, 9,10-very long.

Concerning the representation of the underling movie contents, the proposed video summary achieved an average score, over all the sequences, of 6.9 and an average standard deviation of 1.7, while for the length of the movie we score an average value of 6.1 and an average standard deviation of 1.5.

Therefore, the proposed summary was perceived as preserving *almost entirely the movie contents*, while preserving in general *an appropriate length*. However, the summary was less efficient for movies with a very complex content, when the number of "don't know" answers was important, e.g. 11 for the movie "La Cancion du Microsillon", or when the perception of the movie varies from a user to another, e.g. "Le Moine et le Poisson" (high dispersion of the answers, standard deviation 2.3).

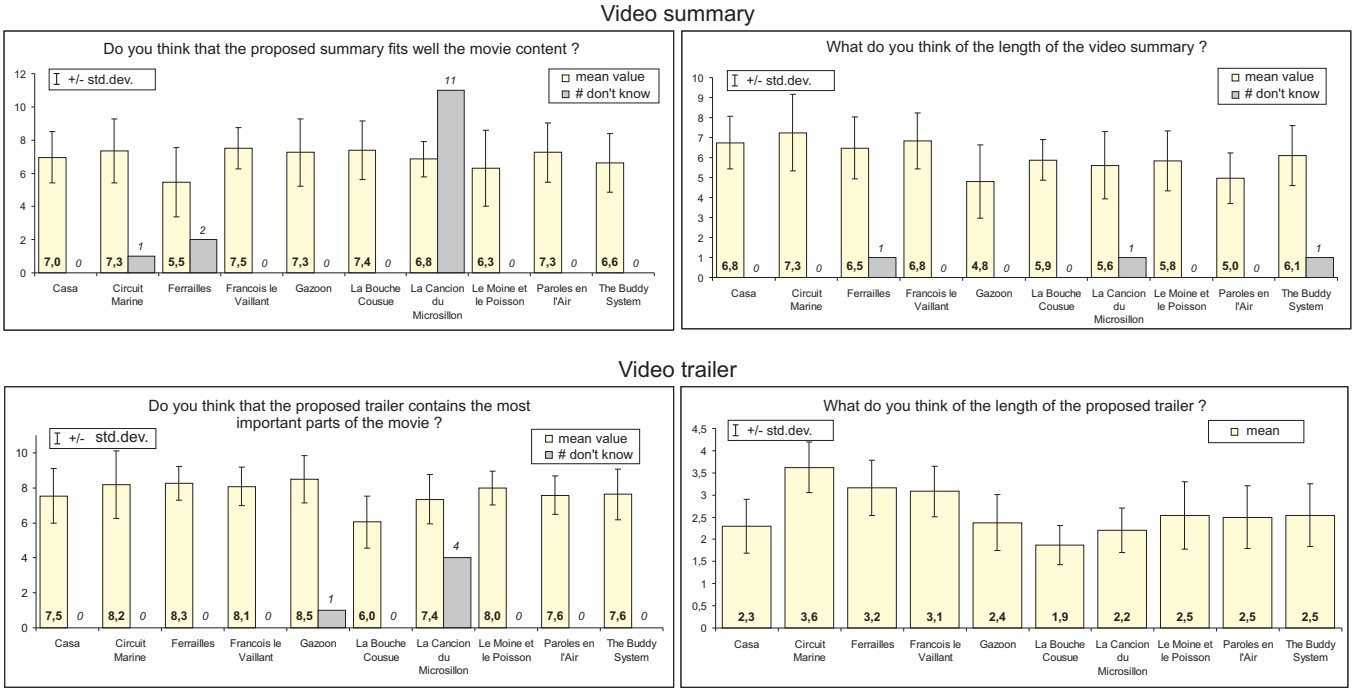

Figure 8: The results of the user evaluation campaign: the oX axis corresponds to the tested movies while the oY axis corresponds to the movie average score, the standard deviation is depicted with vertical segments and gray bars correspond to the number of "don't know" answers.

9.2 Video trailer evaluation

In this case, the user evaluation consisted in answering two questions about the coverage of the movie action parts, as well as concerning the length of the summary, namely:

- for the first question, "*Do you think that the proposed trailer contains the most important parts of the movie ?*", the answers are quantified with a score ranging from 0 to 10, with the following meaning: 0-don't know, 1,2-not at all, 3,4-very few, 5,6-some, 7,8-almost all and 9,10-all of them;

- the second question "*What do you think of the length of the proposed trailer ?*" uses scores from 0 to 4, meaning: 0-very short, 1-short, 2-appropriate, 3-long and 4-very long.

The results are depicted in Figure 8. Overall, the proposed video trailer was perceived as providing *almost all the movie's important parts*, with a global average score, over all the sequences,

of 7.7 and a standard deviation of 1.3. This corresponds to our goal, as video trailers do not aim at providing all the action contents or exciting parts. Compared to the video summary, thanks to the dynamic content, the trailer was naturally more attractive for the viewers, thus the answers are less dispersed (smaller standard deviation) while the number of "don't know" answers is reduced (5 vs. 14).

The trailer length, was considered *appropriate*, with a global average score of 2.6 and a standard deviation of 0.6. However, for movies with a predominant action content, e.g. "François le Vaillant", "Ferrailles", the trailer length tends to be longer (see Figure 8).

In Table 3 we compare the length of the proposed trailer against the original movie and a standard skimming approach which consists of retaining $p\%$ frames from each individual video shot (we take p=25% which is close to the average p value used with the trailer construction, see Section 8.2).

One may notice that the trailer provides a good reduction of the original movie contents, with an average compression around 9:1, while the max value is up to 21:1 (see Table 3). This is done while preserving the movie's most important parts, as confirmed by the previous results. Also, compared to the standard approach, in most of the cases, the trailer is more efficient. The only cases when the trailer approaches the skim length is for some of the movies with a predominant action content, that is a high action ratio (see in Table 3 the movies with an action ratio above 85%), in which the action ratio is defined as the total length of the action clips divided by the total movie length.

10 Conclusions

In this paper we address the video abstraction issue for content-based browsing of animated movies in the framework of automatic video indexing. We propose a unified approach for generating, both static summaries and dynamic video skims, to help in browsing video content, as each type of abstract proved to be efficient in different scenarios.

For a quick browse of the movie visual content, video summaries are the fastest solution, therefore we propose a storyboard-like summary, which follows the movie events by providing each particular movie scene with one key frame. This is carried out at shot level by capturing the shot visual activity with histograms of cumulative inter-frame distances. The number of key frames is adapted to histogram mode distribution and thus fits the shot visual content.

On the other hand, for a "sneak peak" of the movie dynamic action content, we propose a trailer-like video skim, which provides only the most interesting parts of the movie. Our approach is based on highlighting action through the analysis of the movie rhythm (frequency of shot changes) coped with the analysis of shot visual activity. To solve the problem of producing trailers for movies with a more or less static content, for which the notion of trailer is rather ambiguous, as we cannot speak of action or exciting scenes, we adapt the notion of action to each movie content.

The novelty of this work is in the efficiency of this relatively standard approach, when transposed and adapted to the specificity of the animation domain. The performance of our approach has been confirmed through several user studies, directly by the "consumers of the product" (i.e. the end-users) as well as through the manual analysis of the results.

Overall, the proposed abstracts were appreciated as being representative enough for movie contents, while maintaining in general an appropriate length. A good reduction of the original video content is achieved with the video trailer, i.e. average compression ratio 9:1, that is while still preserving most of the interesting parts of the movie. Compared to the video summary, the trailer was naturally more attractive for the viewers, thanks to the dynamic content. One possible drawback, is the summary's length, which, for movies with a high activity content tends to be perceived as being long.

However, one may notice in general the subjectivity of the video abstraction evaluation process. Even involving the user, it is difficult to have an objective measure of the quality of the results. For instance, movies with a complex content resulted in a high dispersion of user answers, while for some other movies, most of the viewers could not decide whether the video abstract is relevant.

Also, was conducting an evaluation campaign can prove to be a very tricky task, and has to be carefully handled, as there are several side effects of such process. For instance, comparing several different skims for the same movie (e.g. proposed trailer vs. standard shot-based skimming approach), is unreliable, as users tend to familiarize with movie contents and no longer differentiate the skims. Also, visualizing movie abstracts for several sequences, as well as the original movies, is time demanding and triggers the user's fatigue. The repeated process, reduces progressively the quality of the evaluation.

Future work mainly consists in enhancing the feature set used for action detection by considering the motion information. At some levels, the visual changes caused by motion are captured with the histograms of cumulative inter-frame distances, but having specific information about fast camera/object motion would be more valuable for the production of the video trailer.

Acknowledgments

This work was partially supported under CNCSIS - Romanian National University Research Council Grant 6/01-10-2007/RP-2 [5] and under Rhône-Alpes region, Research Cluster 2 - LIMA project.

The authors would like to thank CITIA - The City of Moving Images [3] and Folimage Animation Company [4] for providing them access to their animated movie database and for the technical support.

References

[1] Y. Li, T. Zhang, D. Tretter, "An Overview of Video Abstraction Techniques", HP Laboratories, HPL-2001-191, `http://www.hpl.hp.com/techreports/2001/HPL-2001-191.pdf`, 2001.

[2] C.-Z. Zhu, T. Mei, X.-S. Hua, "Video Booklet - Natural Video Browsing", ACM Multimedia, pp.265266, Singapore, 2005.

[3] CITIA - City of Moving Images, `http://www.citia.info/`.

[4] Folimage Animation Company, http://www.folimage.fr/.

[5] B. Ionescu, "Content-Based Semantic Retrieval of Video Documents, Application to Navigation, Research and Automatic Content Abstraction", LAPI - Image Processing and Analysis Laboratory, CNCSIS Project RP-2, http://alpha.imag.pub.ro/VideoIndexingRP2/, 2007.

[6] B.T. Truong, S. Venkatesh, "Video abstraction: A systematic review and classification", ACM Transactions on Multimedia Computing Communications and Applications, 3(1), pp. 3, 2007.

[7] A. D. Doulamis, N. Doulamis and S. Kollias, "Non-sequential video content representation using temporal variation of feature vectors", IEEE Transactions on Consumer Electronics, 46(3), 2000.

[8] J. Ronh, W. Jin, L. Wu, "Key Frame Extraction Using Inter-Shot Information", IEEE International Conference on Multimedia and Expo, 1, Taiwan, 2004.

[9] Z. Li, G. Schuster, A.K. Katsaggelos, B. Gandhi, "Optimal Video Summarization With a Bit Budget Constraint", IEEE International Conference on Image Processing, 1, pp. 617-620, Singapore, 2004.

[10] B. Ionescu, D. Coquin, P. Lambert, V. Buzuloiu, "A Fuzzy Color-Based Approach for Understanding Animated Movies Content in the Indexing Task", Eurasip Journal on Image and Video Processing, special issue on "Color in Image and Video Processing", 1, pp. 20-36, 2008.

[11] H.-C. Lee and S.-D. Kim, "Iterative key frame selection in the rate-constraint environment", Signal Processing: Image Communication, 18, pp. 1-15, 2003.

[12] G. Ciocca, R. Schettini, "Dynamic Storyboards for Video Content Summarization", 8th ACM International Workshop on Multimedia Information Retrieval, pp. 259-268, Santa Barbara, California, 2006.

[13] J.-Q. Ouyang, J.-T. Li, Y.-D. Zhang, "Replay Boundary Detection in Mpeg Compressed Video", International Conference on Machine Learning and Cybernetics, 5, 2003.

[14] A. Hanjalic, H.J. Zhang, "An Integrated Scheme for Automated Video Abstraction Based on Unsupervised Cluster-Validity Analysis", IEEE Transactions on Circuits and Systems for Video Technology, 9(8), 1999.

[15] Z. Xiong, R. Radhakrishnan, A. Divakaran, "Generation of Sports Highlights using Motion Activity in Combination with a Common Audio Feature Extraction Framework, IEEE International Conference on Image Processing, 1, pp. 5-8, Barcelona, Spain, 2003.

[16] F. Coldefy, P. Bouthemy, "Unsupervised Soccer Video Abstraction Based on Pitch, Dominant Color and Camera Motion Analysis", ACM Multimedia, pp. 268-271, New York, 2004.

[17] H. Pan, P. Beek, M. Sezan, "Detection of Slow-Motion Replay Segments in Sports Video for Highlights Generation", IEEE International Conference on Acoustics, Speech and Signal Processing, 3, pp. 1649-1652, Salt Lake City, 2001.

[18] B. Yu, W.-Y. Ma, K. Nahrstedt, H.-J. Zhang, "Video Summarization Based on User Log Enahnced Link Analysis", ACM Multimedia, pp. 382-391, Berkley, 2003.

[19] R. Laganiere, R. Bacco, A. Hocevar, P. Lambert, G. Pays, B. Ionescu, "Video Summarization from Spatio-Temporal Features", ACM International Conference on Multimedia, Trecvid BBC Rushes Summarization Workshop, Vancouver, Canada, 2008.

[20] Animaquid Animated Movie Indexing System, `http://www.annecy.org/home/index.php?Page_ID=44`.

[21] B. Ionescu, D. Coquin, P. Lambert and V. Buzuloiu, "Fuzzy Semantic Action and Color Characterization of Animation Movies in the Video Indexing Task Context", Springer-Verlag LNCS - Lecture Notes in Computer Science, Eds. S. Marchand-Maillet et al., no. 4398, pp. 119-135, 2007.

[22] A.G. Money, H. Agius, "Video Summarisation: A Conceptual Framework and Survey of the State of the Art", International Journal of Visual Communication and Image Representation, 19, pp. 121-143, 2008.

[23] R. Lienhart, S. Pfeiffer, W. Effelsberg, "Handbook of Multimedia Computing", Automatic Trailer Production, NY: CRC Press, pp. 361-378, 1998.

[24] A.F. Smeaton, B. Lehane, N.E. O'Connor, C. Brady, G. Craig, "Automatically Selecting Shots for Action Movie Trailers", ACM International Workshop on Multimedia Information Retrieval, pp. 231-238, Santa Barbara, 2006.

[25] H.W. Chen, J.-H. Kuo, W.-T. Chu, J.-L. Wu, "Action Movies Segmentation and Summarization based on Tempo Analysis", ACM International Workshop on Multimedia Information Retrieval, pp. 251-258, New York, 2004.

[26] Y. Kawai, H. Sumiyoshi, N. Yagi, "Automated Production of TV Program Trailer using Electronic Program Guide", ACM International Conference on Image and Video Retrieval, pp. 49-56, Amsterdam, 2007.

[27] B. Ionescu, P. Lambert, D. Coquin, L. Ott, V. Buzuloiu, "Animation Movies Trailer Computation", ACM Multimedia, Santa Barbara, 2006.

[28] B. Ionescu, V. Buzuloiu, P. Lambert, D. Coquin, "Improved Cut Detection for the Segmentation of Animation Movies", IEEE International Conference on Acoustic, Speech and Signal Processing, Toulouse, France, 2006.

[29] W.A.C. Fernando, C.N. Canagarajah, D.R. Bull, "Fade and Dissolve Detection in Uncompressed and Compressed Video Sequence", IEEE International Conference on Image Processing, Kobe, Japan, pp. 299-303, 1999.

[30] C.-W. Su, H.-Y.M. Liao, H.-R. Tyan, K.-C. Fan, L.-H. Chen, "A Motion-Tolerant Dissolve Detection Algorithm", IEEE Transactions on Multimedia, 7(6), pp. 1106-1113, 2005.

[31] L. Ott, P. Lambert, B. Ionescu, D. Coquin, "Animation Movie Abstraction: Key Frame Adaptative Selection based on Color Histogram Filtering", Computational Color Imaging Workshop at the International Conference on Image Analysis and Processing, Modena, Italy, 2007.

[32] R. Lienhart, "Reliable Transition Detection in Videos: A Survey and Practitioners Guide", International Journal of Image and Graphics, 1(3), pp. 469486, 2001.

[33] Cees G.M. Snoek, M. Worring, "Multimodal Video Indexing: A Review of the State-of-the-art, Multimedia Tools and Applications, 25(1), pp. 535, 2005.

[34] G. Ciocca, R. Schettini, "An Innovative Algorithm for Key Frame Extraction in Video Summarization", Springer Journal of Real-Time Image Processing, 1(1), pp. 69-88, 2006.

[35] J. E. Jackson, "User's Guide to Principal Components", JohnWiley and Sons, New York, NY, USA, 1991.

[36] H. Cheng, Y. Sun, "A Hierarchical Approach to Color Image Segmentation using Homogeneity", IEEE Transactions on Image Processing, 9(12), pp. 2071-2082, 2000.

Table 1: Movie rhythm versus action content.

Movie	Segment [frames]	Length [s]	$\bar{v}_T$
"Hot action"			
François le Vaillant	2961-3443	19	3.51
	9581-10134	22	3.82
	11456-11812	14	3.25
Ferrailles	5303-5444	6	5
	8391-8657	11	3.38
Circuit Marine	7113-7401	11	3.7
The Lyon and the Song	14981-15271	12	2.33
Toy Story	2917-3582	27	2.75
	99962-101090	45	3.84
	101710-102180	19	4.43
Le Moine et le Poisson	6428-6775	14	3.5
"Regular action"			
Toy Story	9901-10596	28	1.75
	27908-28567	26	1.85
	59025-59790	31	1.95
	68230-68869	26	1.92
	76128-77138	40	2.22
François le Vaillant	1561-1957	16	2.14
	7969-8293	13	1.94
	11761-12272	20	1.4
"Low action"			
Le Trop Petit Prince	633-1574	38	0.31
	6945-8091	46	0.37
François le Vaillant	4257-6523	91	0.18
	6898-7683	31	0.38
A Bug's Life	4662-5535	35	0.17
	37209-38769	62	0.62
	66027-67481	58	0.46

Table 2: Action groundtruth.

Action type	"Hot action"	"Regular action"	"Low action"
$E\{\bar{v}_T\}$	3.65	1.97	0.48
$\sigma_{\bar{v}_T}$	0.85	0.29	0.23
interval	2.8-∞	1.67-2.26	0.25-0.71

Movie	Length	Trailer length	Shot-based skim length	# Shots	Action ratio	Compression ratio
"François le Vaillant"	8min56s	**1min25s**	2min15s	164	70%	6:1
"La Bouche Cousue"	2min48s	**16s**	42s	39	52.5%	10:1
"Ferrailles"	6min15s	**1min31s**	1min34s	138	98%	4:1
"Casa"	6min15s	**42s**	1min30s	49	87%	9:1
"Circuit Marine"	5min35s	**55s**	1min22s	125	87%	6:1
"Gazoon"	2min47s	**35s**	42s	31	89%	4:1
"La Cancion du Microsillon"	8min56s	**52s**	2min13s	97	55%	10:1
"Le Moine et le Poisson"	6min	**55s**	1min30s	99	74%	7:1
"Paroles en l'Air"	6min50s	**57s**	1min42s	63	77%	7:1
"The Buddy System"	6min19s	**1min**	1min36s	77	77%	6:1
"A Viagem"	7min32s	**1min**	1min48s	54	71%	8:1
"David"	8min12s	**23s**	1min58s	27	40%	**21:1**
"Greek Tragedy"	6min32s	**24s**	1min36s	29	48%	16:1

Table 3: A comparative study of the archived trailer length.

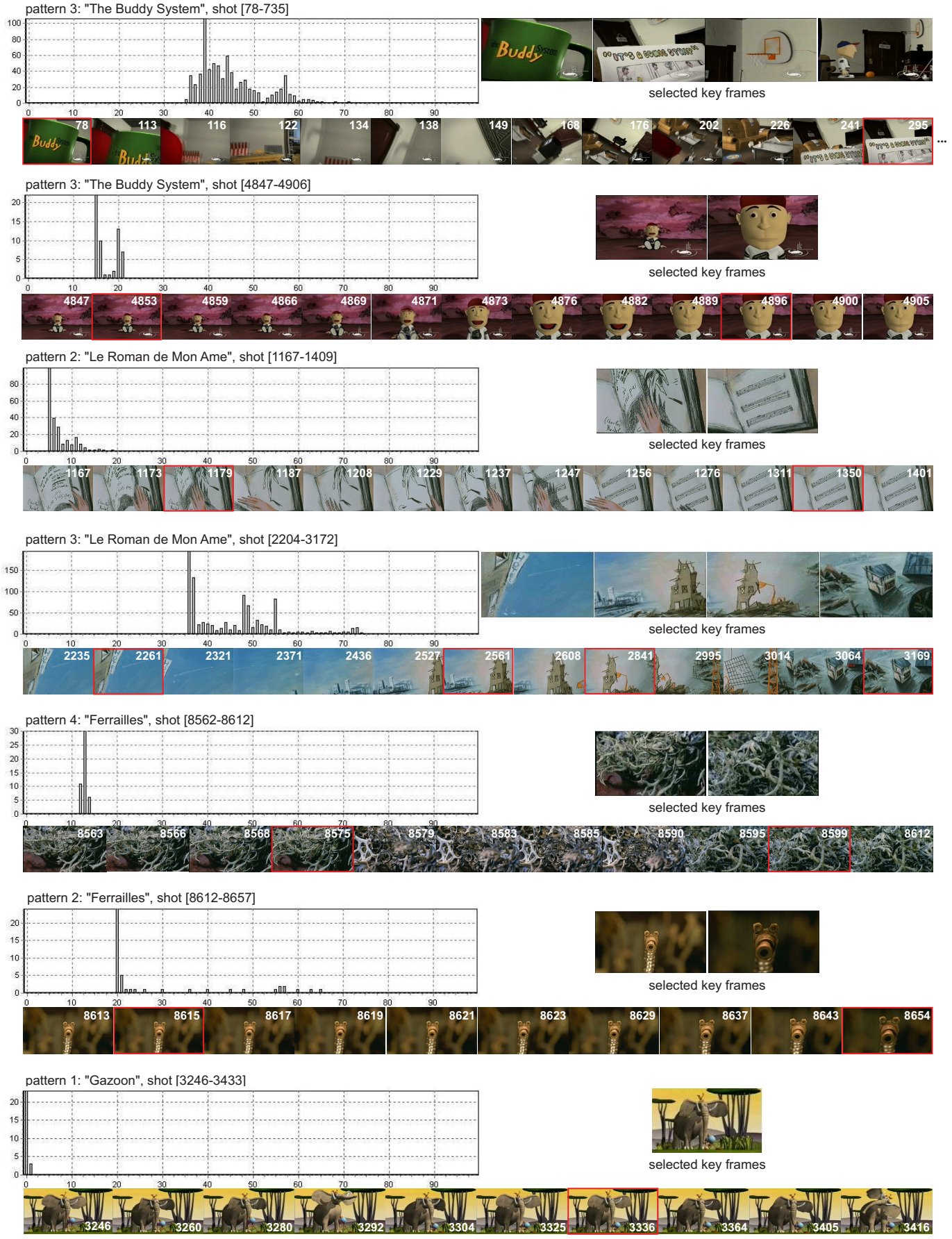

Figure 9: Shot summarization using histograms of cumulative inter-frame distances: oX axis corresponds to histogram bins while oY axis corresponds to histogram value. Each shot is summarized with several representative images for visualization purpose (bottom of the histogram). Selected key frames are marked with red rectangles (detailed on the right side of the histogram).