

A novel video parsing method with improved thresholding*

Alan Hanjalic , Reginald L. Lagendijk , Jan Biemond

Department of Electrical Engineering, Information Theory Group
Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Netherlands
Phone: +31-15-2783084, Fax: +31-15-2781843
e-mail: {alan,inald,biemond}@it.et.tudelft.nl

Keywords: Shot change detection, video libraries, vision, multi-media, recognition

Abstract

Video parsing is one of the steps needed for indexing large volumes of video in digital multimedia libraries. This paper starts with the discussion about some important problems when applying the existing video parsing methods in the practice. These problems are especially related to the user-friendliness of home mass storage systems developed for individual consumers. In an attempt to solve these problems a novel detection method for abrupt shot changes is proposed in the second part of the paper.

1 Introduction

With the continuous increase in capacity of modern storage media, the concept of *digital libraries* becomes technically realizable [1]. In parallel with some other world-wide efforts related to practical realization of such mass storage systems, the European research project SMASH [<http://www-it.et.tudelft.nl/pda/smash>] attempts to show the feasibility of a large capacity home storage device for multimedia data.

Due to the huge amount of stored data in such systems, browsing/query typically takes place on a much smaller data set that, in one way or another, characterizes the stored information [3]. This representative data set we call an *abstract*. When working towards the increase in user-friendliness of

mass storage systems, certain requirements must be set on the abstract making procedure, which provides the material for the actual user-interaction with the system. Especially in case of home storage systems this procedure should be carried out fully automatically, with minimized interaction from the side of the consumer, but still giving an acceptable quality of the abstract. Another increase in user-friendliness can be obtained by speeding-up this procedure, i.e. by performing all necessary steps at recording time. Furthermore, the implemented algorithms should be sequence-independent.

This paper deals with the first step in making an abstract for video data: *video parsing*. Due to our activities within the SMASH project, we are - among other topics - interested in research towards the video parsing methods where the mentioned requirements are fulfilled. In case of video parsing, especially the sequence processing at recording time (“on-the-fly”) and the sequence-independence of the applied parsing algorithm are required.

In the following sections, after discussion of existing video parsing approaches we propose an automated detection method for abrupt shot changes which uses locally computed thresholds based on a statistical model for frame-to-frame differences. It enables the detection directly on the incoming video stream, and has similar performance for any video sequence. This performance can be estimated using statistical parameters of the model. In section 5 some conclusions related to the proposed method can be found.

2 General problem of video parsing

* This work was supported in part by the EU ACTS program under the contract AC018: SMASH (Storage for Multimedia Applications Systems in the Home)

Video can be seen as an ordered collection of unbroken and continuous series of frames, called shots. The first step in the process of creating a video abstract is the detection of these shots in the video stream. This temporal segmentation process is also called video parsing.

Video parsing is based on measuring changes between consecutive frames. In measuring these changes only real *content-changes* should be taken into account. For instance, small movements of objects on a static background, camera movements, focal length changes, or changes in luminance should not be registered as relevant changes [6]. To measure relevant contents changes we need an appropriate feature describing the frame content and a metric to evaluate changes of that feature. This results in a frame-to-frame difference time function $FFD(k)$. The way of obtaining the $FFD(k)$ in this paper is explained in section 4.1.

Using the $FFD(k)$ we can detect shot changes. Abrupt shot changes can be detected as sharp peaks in the FFD curve, while gradual transitions between shots require more elaborate detection mechanisms. Existing detection approaches do not comply with requirements needed for consumer home storage systems. This is the case even for detecting sharp peaks in the FFD curve. In the following we therefore concentrate on the problem of abrupt shot changes detection.

3 Current approaches for detection of abrupt shot changes

Only two thresholding approaches for detecting sharp transitions in the $FFD(k)$ function can be found in the literature. In [5] the use of a fixed threshold for the entire video sequence is proposed. This threshold is determined under the assumption that $FFD(k)$ is Gaussian distributed, except for those values resulting from shot transitions or camera movements. The drawback of this approach is that statistical information of the entire $FFD(k)$ function is needed, implying that the entire video sequence has been stored. Another problem when using a global threshold appears in cases where a very distinguishable break-peak can be observed in one stationary part of the sequence, but whose height is similar to FFD values along a high-action shot in an other portion of the sequence. An example for this can be seen in Figure 1.

In [2] a local threshold is determined within a sliding window containing several last computed FFD values. The window is determined such that it cannot contain more than one shot change. For the case of sharp shot changes the middle window point is

always checked (1) to be the maximum and (2) to be X time higher than the second largest FFD value in the window. Computing the threshold locally, i.e. only using the information from a short temporal segment has several important advantages:

- the threshold function changes much more rapidly, following the function $FFD(k)$. This can enable the proper detection even in cases like in Figure 1 without causing false alarms.
- It is suitable for on-the-fly video parsing since only the current information about the sequence is needed.
- It possesses the “forgetting” property, i.e. the detection performance is not influenced by missed and false detections which happened before.

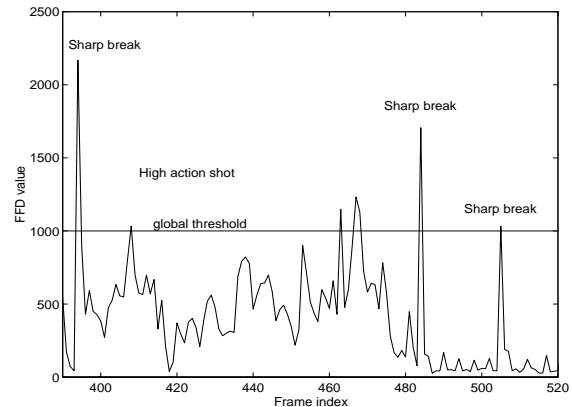


Figure 1: Problems by using global threshold

However, a disadvantage of the method proposed in [2] is that the overall performance of the system is rather sensitive to the setting of the parameter X , which makes it unsuitable for our purposes.

4 Novel approach

We propose a method for detecting sharp shot changes using all advantages of local thresholding but being far more sequence-independent than the method from [2], for any given parameter value. For this purpose we introduce a general statistical model for the FFD curve. Such a model enables us to work with *detection probabilities*, as shown in section 4.3.1.

4.1 Features and metrics

Popular features to measure global frame contents are the color and intensity histograms [2, 3, 4, 6]. We have used YUV histograms with 64 bins for the chrominance and 32 bins for the luminance information. As metric we used the sum of the absolute histogram differences to obtain FFD values:

$$d_{YUV}(k, k-1) = \sum_{j=Y,U,V} \sum_i |h_k^j(i) - h_{k-1}^j(i)| \quad (1)$$

4.2 Statistical model for the $FFD(k)$ curve

In [5] it was observed that $FFD(k)$ (there obtained by comparing color code histograms) can be regarded as a realization of an uncorrelated Gaussian process, if no shot change or motion is present. This observation we like to extend to any other temporal segment with *uniform* content development, independent of the present amount of action. Within a single shot the $FFD(k)$ can then be modeled either as a single uncorrelated Gaussian process or as a temporal concatenation of multiple uncorrelated Gaussian processes. Shots themselves are separated by individual large-valued outliers, or peaks.

As an illustration of this, consider a camera following a soccer player. If initially the camera is stationary, frame-to-frame differences are caused by the moving of the soccer player within the frame window. The resulting variations of $FFD(k)$ will be around one mean value, and are the result of a large number of small differences. The central limit theorem tells us that the resulting summed difference will be approximately Gaussian. If the camera starts panning to follow the running player, the mean and variance of $FFD(k)$ will shift to other values. If then another camera view is selected, a shot change results, yielding a peak in $FFD(k)$ followed by FFD values around another mean and with another variance. Another example would be a still camera taking from the close distance the running crowd during the marathon race. Due to a high amount of action, FFD values might be large, but they remain grouped around a (large) mean value and having a (large) variance.

Figure 2a illustrates the FFD curve of a high-action shot. At the onset and at the end of the shot the peaks due to the shot change can be seen. Figure 2b shows the $FFD(k)$ histogram and a Gaussian curve derived from the mean and standard deviation estimated from the FFD values.

From the many results published in the literature and from our own experiences we conclude that a

statistical model for the $FFD(k)$ function should at least show the following properties:

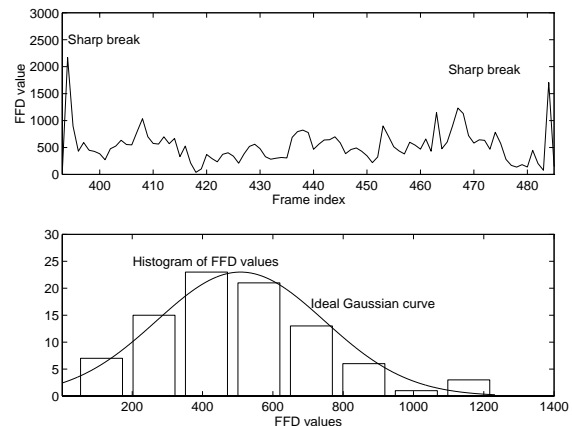


Figure 2a: Frame-to-frame differences of a shot with strong but uniform action

Figure 2b: Distribution of FFD values along the shot and ideal Gaussian curve derived from the mean value and standard deviation

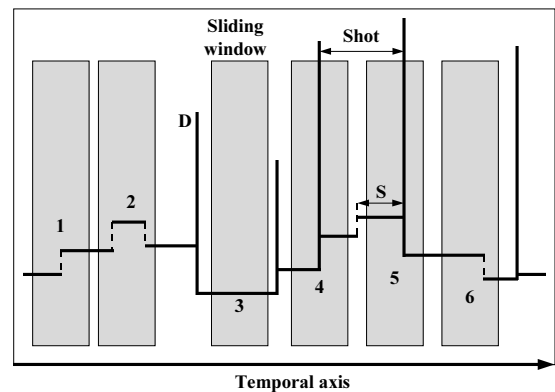


Figure 3: Illustration of a statistically modeled fictive $FFD(k)$ curve with some possible positions (numerated) of the sliding window.

- $FFD(k)$ has an underlying two state model, with state “S” for being within a Gaussian shot segment, and with state “D” for shot changes. A state “S” can be followed by another state “S” or by state “D”. State “D” is always followed by state “S”;
- Each state “S” has three parameters determining the process that generates $FFD(k)$ in that state, namely: the duration of the state L , the mean and variance of the uncorrelated Gaussian process $FFD(k)$ in that state;
- State “D” has duration 1;

In Figure 3 a statistical model of a fictive $FFD(k)$ function can be seen, with each Gaussian segment “S” represented by its mean value. In principle one could now develop a recursive detector that determines the states from which the observed $FFD(k)$ values originate. This would automatically lead to a shot change detector. In the following, however, we will extend the sliding window thresholding approach from [2] to take the above model for $FFD(k)$ into account.

4.3 Sliding window thresholding

In the sliding window approach, detection takes place on the center value of the window [2]. The length of the sliding window is selected such that it is highly unlikely that two shot changes occur within the window. Then, at a frame-rate of n frames/second the length of the window can be chosen as not larger than $2n-1$, if it is assumed that two shot changes within one second are rare. Another issue by defining the window dimension is the minimal length of each “S” state. We assume that this minimal value of L corresponds to one half of the window length.

Detection of a shot change requires the detection of an outlier in the window. This detection takes place in two steps, namely:

- the center value in the window must be the largest FFD value in the window;
- the center value must be larger than the locally computed threshold.

The central question is still what the threshold value in step 2 should be. To determine this threshold, let us consider the various situations that can occur on the basis of the statistical model for $FFD(k)$ described in the previous section. Due to the step 1, there is no need to compute the threshold and start the detection procedure if the center window value is not the maximum of the window. Therefore, we are interested mainly in cases where this condition is, or may be satisfied. Examples of these cases can be seen in Figure 3 and can be divided in two classes:

- *Class 1*: all cases with two “S” states surrounding the middle of the window (positions 4, 1 and 3)
- *Class 2*: all other cases than in *Class 1* where the middle window point is only by chance the maximum (possible in positions 2 and 6).

4.3.1 Discussion on Class 1

Assuming that P is the given probability for false detection of a shot change, the threshold T for the sliding window in this case can be determined as the solution of the integral equation

$$P = 0.5 * \int_T^{\infty} (p_{left}(z) + p_{right}(z)) dz \quad (2a)$$

with $p_{left/right}(z)$ being the function of the Gaussian distribution for the left/right side of the window. P is the only parameter to be inserted manually. It determines the probability that a FFD value belonging to a “S” state is higher than the computed threshold, which would lead to false detection. Also an approximate solution for this threshold can be used, where only the *dominant* integral contribution in (2a) is taken into account.

$$P = \int_T^{\infty} p_{dom}(z) dz \quad (2b)$$

The threshold T is then obtain as the solution of equation (2b) as

$$T = \mu + \alpha * \sigma \quad (3)$$

with the parameter α corresponding to the given probability P and the statistical parameters taken for the normalized dominant Gaussian curve $p_{dom}(z)$. We found the detection performance for this approximate threshold value to be fully acceptable, compared with the exactly computed threshold. More detailed discussion dealing with missed detections can be found in section 4.4.

4.3.2 Discussion on Class 2

The main problem in cases belonging to this class comes from the attempt to determine the threshold, as described in section 4.3.1, whereby at least on one window side a transition between two Gaussian segments can be found. For that window side the usage of introduced formulas is expected to give unpredictable results, which may lead to false detections for these window positions.

4.4 Discussion on the overall detection performance

Although the cases from the *Class 2* can be understood as a source of possible problems in the detection, we found that their influence on the overall detection performance is not large, apart

from a slight increase in the false alarm rate (one false alarm in experimental results, section 4.5).

This leaves us the well modeled *Class 1*, as described in section 4.3.1, to be discussed in more details.

The performance of a detection algorithm is evaluated by estimating the probability for false and missed detections. The probability for false detections is in *Class 1* directly controlled by the parameter P . If the threshold computation (3) is used, each given value for P results in an appropriate value for α . Due to unknown general distribution of values belonging to “D” states, it is difficult to formulate the similar sequence-independent probability for missed detection. However, if the peaks in the $FFD(k)$ function are distinguishable enough from all other FFD values in a sequence, the distribution of “D” states is also strongly separated from any Gaussian distribution of “S” states. A “universal” threshold providing an acceptable performance for both false alarms and missed detections, can be set in the ideal case in the area in-between, where both distributions go towards zero. A near-to-ideal case can be approached by using well selected features and metrics for measuring FFD values. Using this analysis and metric (1), we obtained acceptable results for α between 5 and 6.

Sequence	Length (frames)	Shot Changes	Missed Detections	False Alarms
Movie 1	12400	71	1	0
Movie 2	8000	46	1	0
Documentary video	5000	13	0	1

Table 1: Results of detecting abrupt shot changes using the proposed statistical algorithm ($\alpha=5$)

In the method from [2] a proper choice for the parameter X can be made to optimize the detection for a limited set of test-sequences, but no conclusions about the performance can be made if any other sequence is taken, not belonging to that set. Therefore, it cannot provide the generality and robustness of the detection performance, required for fully automated systems, as it is possible with our novel approach for any given parameter P .

One possible weak point of the proposed method we see in the fact that the number of FFD values within the window may not be large enough for building a reliable statistics. Therefore we must talk here about *approximated* statistical parameters of each Gaussian segment of the sequence.

4.5 Experimental results

The performance of the detection algorithm has been evaluated on “real-movie” sequences with 80x64 subsampled frames. Sequences used in tests have variable global content developments, and contain stationary as well as high-action shots. Also the cases with different content developments within one and the same shot were not rare. Tests have been made using the approximate threshold computation described in section 4.3.1. In view of the analysis from the last section, we worked only with one “universal” threshold, for both false alarms and missed detections.

To demonstrate the performance of the approach we chose $\alpha=5$. From the total number of sharp shot changes of 130 in all sequences together, 2 missed detections (1.5%) and 1 false alarm (0.8%) were registered. Each of these three detection mistakes happened in different sequences, which leads to a similar detection performance in each of them for the given value of α . The window length was chosen as 21. Table 1 shows the results for each sequence separately.

5 Conclusions

In this paper we introduced a novel approach for performing the video parsing process by taking into account the requirements for user-friendliness mentioned in the introduction to this paper. This makes the method applicable especially in home storage systems developed for individual consumer, like SMASH. In the proposed method, good results are obtained using the statistical model for generalizing the performance of the shot change detection procedure. Our intention in further research is to make the model be more sophisticated and extend it for the detection of gradual transitions.

6 References

- [1] *COMPUTER* - IEEE Computer Magazine, Vol. 29, Issue 5, May 1996.
- [2] B. Yeo, B. Liu.: “Rapid Scene Analysis on Compressed Video”, IEEE Transactions on Circuits and Systems for Video Technology, Vol.5, No.6, December 1995.
- [3] B. Furht, S.W. Smoliar, H. Zhang: “Video and Image Processing in Multimedia Systems”, Kluwer Academic Publishers, 1995
- [4] G. Ahanger, T.D.C. Little: “A survey of Technologies for Parsing and Indexing Digital Video”, Journal of Visual Communications and Image Representations, vol.7., No. 1, pp. 28-43,1996
- [5] H. Zhang, A. Kankanhalli, S.W. Smoliar: “Automatic partitioning of full-motion video”, Multimedia Systems, Vol.1, pp 10-28, 1993

- [6] I.K. Sethi, N. Patel: "A Statistical Approach to Scene Change Detection", in Proceedings of SPIE, Vol. 2420, pp.329-337, 1995
- [7] A. Hanjalic, M. Ceccarelli, R.L. Lagendijk, J. Biemond: "Automation of system enabling search on stored video data", in Proceedings of SPIE Storage and Retrieval for Image and Video Databases V, Vol. 3022, San Jose, 1997

