Multimed Tools Appl (2006) 30: 273–288
DOI 10.1007/s11042-006-0026-2

Information-theoretic temporal segmentation of video and applications: multiscale keyframes selection and shot boundaries detection

Bruno Janvier · Eric Bruno · Thierry Pun · Stéphane Marchand-Maillet

Published online: 7 September 2006

Abstract The first step in the analysis of video content is the partitioning of a long video sequence into short homogeneous temporal segments. The homogeneity property ensures that the segments are taken by a single camera and represent a continuous action in time and space. These segments can then be used as atomic temporal components for higher level analysis like browsing, classification, indexing and retrieval. The novelty of our approach is to use color information to partition the video into segments dynamically homogeneous using a criterion inspired by compact coding theory. We perform an information-based segmentation using a Minimum Message Length (MML) criterion and minimization by a Dynamic Programming Algorithm (DPA). We show that our method is efficient and robust to detect all types of transitions in a generic manner. A specific detector for each type of transition of interest therefore becomes unnecessary. We illustrate our technique by two applications: a multiscale keyframe selection and a generic shot boundaries detection.

Keywords Content-based video analysis · Temporal segmentation · Keyframe selection · Detection of shot boundaries

1 Introduction

The increasing amount of video documents produced every day creates a new need for the management and retrieval in multimedia information systems. The first step to achieve in this research area is the temporal partitioning of any video in sub-sequences that represent a continuous action in time and space for the purpose of further indexing.

The problem of shot-boundary detection has been tackled by many computer vision scientists without being completely solved. In the survey by Koprinska and Carrato [4], a number of techniques of temporal segmentation of uncompressed or compressed video are described. Many methods are related to the detection of discontinuities using pairwise pixel,

B. Janvier (✉) · E. Bruno · T. Pun · S. Marchand-Maillet
Viper Group, Computer Vision and Multimedia Laboratory, Université de Genéve, Geneva, Switzerland
e-mail: janvier@cui.unige.ch

block based or histogram comparisons. In the compressed domain, DCT coefficients are used instead of pixel values.

Whereas the detection of a discontinuous camera cut (hardcut) in a video sequence is relatively easy, a transition can also be gradual and due to a dissolve, fade or wipe special-effect transition. A transition in the action may also be due to the fact that the camera shows one thing at a given time and a completely different thing at another time; the transition can simply be a rotation or a zoom of the camera. Especially for indexing purposes, it is important to detect these events. There are plenty of different types of transitions that do not show any abrupt discontinuities (due to the presence of special effects or not) and their detection is therefore difficult as shown in the survey of Kasturi et al. [1]. It is proposed in many articles to design a specific detector for each type of special-effect transitions [3, 5]. Here, we rather depart from this solution in order to avoid ad hoc techniques.

The novelty of our approach is to use color information to chop the video into dynamically homogeneous segments using a criterion inspired by compact coding theory. An information-based segmentation using a Minimum Message Length (MML) criterion will be applied to partition the video into segments where the evolution is homogeneous by taking into account all the available information. In the literature, many clustering-based segmentation methods exist that use, for example, hierarchical clustering of frame dissimilarity. The approach we have chosen is different. The segments are inferred in order to maximize locally the homogeneity of the evolution but also to minimize globally the complexity of the partitioning using a Dynamic Programming algorithm. The optimization process is global and therefore more satisfactory than greedy or agglomerative strategies.

This framework provides atomic temporal components for higher level content-based video analysis. Many problems are simplified, we will present two applications. The first application is about video overviewing. A multiscale segmentation and keyframe selection will be performed in order to let the user of the application interactively adjust the degree of coarseness of the keyframe representation. The second application presented deals with the detection of shot boundaries. In television or cinema, the base unit is the shot. Shots are separated by abrupt or gradual transitions. We detail a simple and generic technique to detect these transitions disregarding the kinds of special effects involved.

2 Dissimilarity profile

In order to perform a temporal segmentation of the video stream, we need a distance measure between two successive frames. The analysis will be done on the resulting temporal profile of the frame-by-frame distances.

At the very least, the similarity measure between two frames should satisfy the following properties:

- it should be stable with respect to changes that are common during a segment representing a continuous action in time and space such as small affine transformations, lighting changes, deformations, appearance of objects, etc.
- it should give an accurate quantitative information about the amount of change that has taken place.

The color histogram has proven to be a very stable representation in the content-based image retrieval research field. The distribution of color is invariant and stable for frames representing a similar content. We will compute the histogram in the YUV color space, because it gives the best performance/speed ratio when dealing with MPEG video streams.

The Jeffrey divergence is used to measure the distance between the color histograms. It measures how compactly one histogram can be coded using the other as a codebook and gives better quantitative results in our experiments than the L_1, L_2 or chi-square metrics. If H_i and H_j are two histograms containing N bins, the Jeffrey divergence between H_i and H_j is defined by:

$$D_{col}(i,j) = \sum_{k=1}^{N} (H_i(k)\ \log\left(\frac{H_i(k)}{m(k)}\right) + H_j(k)\ \log\left(\frac{H_j(k)}{m(k)}\right)) \tag{1}$$

where $m(k) = \frac{H_i(k)+H_j(k)}{2}$.

By computing the Jeffrey divergence for the pairwise computation of histogram differences for the complete video stream, we obtain a frame-by-frame dissimilarity profile $Diss_{info}$ that will be used in the next sections:

$$Diss_{info}(i) = D_{col}(i, i+1) \tag{2}$$

Note that, in our framework, video information is abstracted by its features. In this respect, it is possible to replace color information, for example by motion or sound, and get a partitioning that will hold a different interpretation than that obtained with the methods that will be presented next. The problem of deriving a set of features that leads to a potential semantic interpretation of the content is out of the scope of this article.

3 Information-based partitioning of ordered data

We present here how the video is partitioned into a series of homogeneous segments. We first define and model what we mean by "homogeneous." Then, we will present a criterion based on compact coding theory that we will minimize in order to infer the number and the location of change-points within the video document. Finally, we also give simplifying heuristics chosen in order to make the algorithm more efficient.

3.1 Properties and modelling of the color dissimilarity profile

The partitioning of our video is done by considering that a video segment has been generated by a given model. The choice of the model is constrained by the following criteria:

- it should be able to fit the data during a homogeneous segment;
- it should show an excellent detection performance in order to capture when dynamic of the video has significantly changed: the model should not fit discontinuities nor any major changes in the temporal evolution;
- it should be generic enough so that it stays valid for any type of video document.

We will use the cumulative sum of the dissimilarity profile for the information-based segmentation because it measures the *trend* defined as the accumulated effect of the fluctuations of a time series.

If the evolution of the colors is homogeneous and the frame-by-frame dissimilarity is roughly constant, the trend is expected to have a linear behavior.

The model that we will use for a segment is thus:

$$y_m^{\theta}(t) = a_1 t + a_0 + e_t \tag{3}$$

The additive error terms, e_t, are assumed to be *i.i.d* and the error density $N(0,\sigma)$ for unknown σ. We use least square estimates of the linear coefficients.

This model is interesting because the grouping by similarity will take into account the static (parameter a_0), but also the dynamic properties (slope a_1 and variance σ) of the color content of the video.

The trend of the whole video will then be modelled as a changing linear regression model with piecewise constant parameters. We need to estimate the number of segments G and the sequence of changing points $s = (s_1, ..., s_G)$.

3.2 Partitioning

The segmentation problem is about finding the partitioning that best explains the data assuming a model $y_m^{\theta}(t)$ with different parameters $\theta = (a_0, a_1, \sigma)$ in each segment.

Recently, a Minimum Message Length (MML) criterion has been used by Fitzgibbon et al. [7] to infer the number of segments and the location of the cut-points from univariate temporal data using Fisher's DPA. The MML criterion has been experimentally proven by the authors to be more powerful to accurately locate the boundaries of the segments than other criteria such as the Minimum Description Length (MDL), the Bayesian Information Criteria (BIC) or Akaike's Information Criteria (AIC) . The MML is based on the compact coding theory [9]. The idea is that the best explanation of the data is the one that provides the briefest encoding of a two-part message. The first part contains the information about the statistical model while the second part contains the remaining information needed about the data assuming the model. This is a quantification of the trade-off between the model complexity and the goodness of fit. The idea is that the partitioning to be preferred is the one that best fits the data using a model as simple as possible. The code length of the messages are computed using Shannon's theory where the length of the string coding an event E in an optimally efficient code is given by $-\log(p(E))$.

The message length used to calculate the expected length of a message which transmits the model and the data of the jth segment containing the time series $y = (y_1, ..., y_n)$ can be approximated according to [8] by:

$$ML(\theta)_j = C(\theta)_j + D(\theta)_j \tag{4}$$

where $C(\theta)_j$ is a penalty term and $D(\theta)_j$ is the data fitting term.

$D(\theta)_j$ corresponds to the ML (Maximum-likelihood) estimator for i.i.d Gaussian errors. The minus log-likelihood is minimized when the model best fits the data. $D(\theta)_j$ corresponds to the code length of specifying the data assuming the model.

$$D(\theta)_j = n \log(\sqrt{2\pi}\sigma_j) + \frac{1}{\sigma_j^2}\sum_{t=1}^{n}(y_t - a_{1j}t - a_{0j})^2 \tag{5}$$

As it stands, this quantity will be minimized for the degenerated solution where each sample is a an independent segment. There is therefore a need for a second term for our solution to follow the parsimonious principle.

$C(\theta)_j$ is a penalty term that takes into account *a priori* information P_r and a code length on the cost of specifying the model on the given set of data of length n.

This code length is related to the uncertainty of the estimation of each of the model parameters. The standard error in the estimated variance is $\pm\frac{1}{\sqrt{n-2}}$ and the associated code length is $\frac{1}{2}\log(n-2)$ assuming an uniform distribution. The standard error in the estimated

linear coefficients is $\pm\frac{\sigma}{\sqrt{n-2}}$ with an associated code length of $\frac{1}{2}\log(n-2) - \log(\sigma)$ for each coefficient.

$$C(\theta)_j = -\log(P_r) + \frac{3}{2}\log(n-2) - 2\log(\sigma) \tag{6}$$

P_r is a prior information that we will design in order to meet our requirements. The *a priori* information will depend on the length of the segment to penalize the creation of too small partitions.

$$P_r = \sum_{w=0}^{\lambda(k)} \frac{\alpha^w}{w!} e^{-\alpha}. \tag{7}$$

The parameter α is chosen such that the prior reaches a non-informative value after a given number of frames; we chose 5 for example. The associated code length is the negative log of the probability.

The partitioning $s = (s_1, ..., s_G)$ containing G segments that maximizes the homogeneity of the data according to the model y_m^θ is also the one that minimizes the total message length:

$$ML_{total} = \log^*(G) + \log\left(\binom{K-1}{G-1}\right) + \sum_{j=1}^{G} ML(\theta)_j \tag{8}$$

where G is the number of partitions and K is the total number of frames of the video, $\log^*(G)$ and $\log\left(\binom{K-1}{G-1}\right)$ are the code length needed to specify the number of segments and which particular partitioning has been chosen assuming that all of them had the same probability.

3.3 Minimization

The problem is now to conduct the optimization in order to get the best partitioning of the ordered set of K numbers into G contiguous groups. This is a combinatorial problem and there are $\binom{K-1}{G-1}$ possibilities to explore.

This problem has been solved in polynomial time by Fisher [6] using a Dynamic Programming Algorithm (DPA). The search algorithm is based on the optimality principle that states that in an optimal sequence of decisions, each subsequence must also be optimal. The time complexity of the DPA is reduced because the optimal solution is a combination of optimal solutions of subinstances. For a set of K numbers and a maximum number of groups $G_{\max}$, the time complexity is $O(G_{\max}.K^2)$.

The MML/DPA strategy presents two very interesting advantages over other segmentation techniques like agglomerative or greedy clustering strategies:

- Global approach: every possible partitioning is taken into account during the minimization process and there is no risk to end in a local minima.
- Parameter-free: no threshold is needed to stop the clustering process and no need to specify the final number of partitions. This is theoretically more satisfactory.

The number G of partitions and the locations of the boundaries are then computed, and we know that this partitioning and this number of partitions will maximize the homogeneity of the data in each partition according to our model. However, the computational complexity is still too high to analyse globally entire real life videos. We will use simplifying

assumptions in order to restrict the search when necessary and to make it as efficient as possible.

3.4 Restricting the search for the solution

In practice, we use two simple ideas in order to significantly speed-up our segmentation by reducing the number K involved in the computational complexity of the Dynamic Programming algorithm. The main idea is that we should perform an information-based segmentation only when necessary, between two abrupt transitions and only if something is going on. It makes it usable for partitioning of large video collections.

3.5 Hardcut detection

An abrupt transition between two different shots is relatively easy to detect in the dissimilarity profile. It is likely to correspond to a strong peak. The hardcut detection can be formulated as a binary-hypothesis test problem. We introduce two hypotheses:

- Hypothesis S: there is an abrupt transition present between frames k and $k+1$
- Hypothesis $\overline{S}$: there is no transition present between frames k and $k+1$

The test can fail if we make a false detection (i.e., S is chosen when $\overline{S}$ is true) or a missed detection (i.e., $\overline{S}$ is chosen when S is true).

A well-known result in hypothesis testing is that the following decision rule is equivalent to the minimum risk of error:

$$\frac{p(z|S)}{p(z|\overline{S})} \underset{S}{\overset{\overline{S}}{\lessgtr}} \frac{1-P_k(S)}{P_k(S)}. \tag{9}$$

In the recent paper from A. Hanjalic [2], the likelihood functions $p(z|S)$ and $p(z|\overline{S})$ and $P_k(S)$, the probability for the validity of S, have been specifically modelled for video hardcut detection. We follow a similar modelling.

As we know for sure that these strong transitions will be present in our final solution when using the DPA, the minimization can then only be performed within these hardcuts. It makes our optimization less dependent on the total length of the document, but simply on the typical length separing two hardcuts. For a video containing K frames and containing X sub-sequences separated by abrupt transitions of length $(K_1, ..., K_X)$, the time complexity is reduced from $O(G_{\max}K^2)$ to:

$$\sum_{x=1}^{X} O(G_{\max}K_x^2) \tag{10}$$

with each $K_x \ll K$.

3.5.1 Greedy grouping

The second preprocessing uses the fact that it seems unnecessary to perform our minimization when it is obvious that nothing happens in the video stream.

We will group together sub-sequences of the video where nothing significant is happening using a very sensitive greedy algorithm and a threshold. Groups of "still" frames will be

regarded as one frame. A significant speed-up is then obtained by starting the minimization of our criteria on a new set of K_{over} sub-sequences such that $K_{over} \ll K_x$. The oversegmentation is found in a very sensitive way such that it does not reduce the optimality of the segmentation, but avoid not useful computations.

4 Applications

Using the algorithm presented, some knowledge about the structure of the raw video has emerged. In this section, we use this temporal segmentation for two applications.

An application of video overviewing is presented using a multiscale keyframe selection technique. The coarseness of the video overview is chosen by the user who has the possibility to adjust a scale parameter.

The second application presented is the detection of shot boundaries. We show a simple and generic technique to detect abrupt or gradual transitions separating shots irrespective of the kinds of special effects involved.

4.1 Multiscale keyframe selection

For video overviewing, we would like to let the user choose the level of accuracy of the keyframe representation of the video. We perform a multiscale temporal segmentation of the video followed by an informative keyframe selection that has the property to be persistent through different scales.

4.1.1 Multiscale segmentation

In a similar manner as the Scale-Sets representation of images of Guigues et al. [10], we build a hierarchy of coarser and coarser segmentation of the video by grouping contiguous segments.

Considering a partition a, a generalization of the criteria of the Eq. 4 is to add a real and positive parameter λ that we will call the scale parameter such that:

$$ML_a = \lambda C_a + D_a \tag{11}$$

As λ is increasing, we constrain more and more the penalty term of the criteria, the number of found partitions G is then decreasing. We thus obtain a progressive simplification of the segmentation until only one segment remains. Hence, λ can be seen as a scale parameter as in regularization algorithms.

We will build a binary tree to represent the progressive merging of contiguous segments when the scale λ grows to infinity. Considering two contiguous partitions a and b, the scale of appearance of a node x in the tree to group these partitions together is defined as λ^+ which is the unique solution of the affine function:

$$ML_x(\lambda^+) = ML_a(\lambda^+) + ML_b(\lambda^+) \tag{12}$$

$$\lambda^+ = \frac{D_a + D_b + D_x}{C_a + C_b + C_x} \tag{13}$$

Considering all the partitions obtained for the complete video with $\lambda = 1$, we compute the scales of appearance of all possible grouping of contiguous segments and choose to group

together the pair of segments having the minimum scale of appearance. We iterate with the newly formed set of partitions until only one segment remains. The hierarchy is progressively built by augmenting the regularization constraint and we get at each scale the partitioning that minimizes globally our criterion. We show an example of such a tree in figure 1.

4.1.2 Keyframe selection

In our framework, the most informative level is found by choosing one keyframe per homogeneous partition for $\lambda = 1$. Considering every homogeneous partition, the keyframe is selected as the frame having a color histogram which is the centroid of the segment.

When the scale parameter λ grows, we decimate our set of keyframes by choosing the most informative ones. For every merging of two homogeneous partitions, we choose the keyframe that is the closest to the centroid of the newly merged segment. This way guarantees a persistency of the keyframes throughout the different scales.

There are many further possible ways to use the hierarchy we have built. The user can interactively change the scale parameter and see the progressive simplification of the keyframe representation of the video. The user may also specify the number of keyframes K (s) he wants to see; the associated scale parameter λ can then be deduced.

The different overviews obtained at different scales are interesting because the segments that have the most similar dynamic behavior (e.g., measured by the slope and the variance of the linear model) will be grouped together first. It does also exhibit interesting semantic

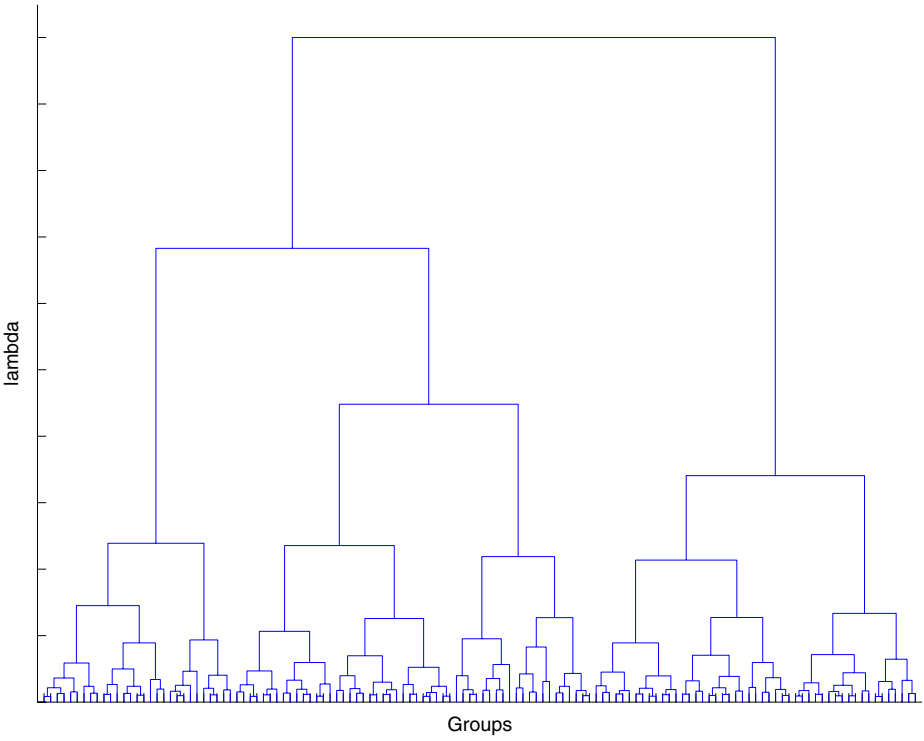

Fig. 1 Binary tree showing the hierarchical grouping of contiguous segments when λ grows

properties. For example, in news videos, the static studio settings with the anchor person segments will be unlikely to be merged together with dynamic outdoor segments as can be seen in figure 5. The user may skip several levels of details, but still preserve the global structure of the news report with the successive apparitions of the anchor person.

4.2 Detection of shot boundaries

4.2.1 Problem statement

The temporal segmentation into homogeneous segments already disclosed important information about the structure of the video, but there are still too many segments in comparison to the number of shots that correspond to the reverse-engineering of the video production process. It is important to make it correctly not to negatively influence higher-level analysis like video summarization.

Within our framework, the detection of shot boundaries is reduced to the merging of the segments that are not due to a transition between shots. We essentially face two kinds of artefacts: signal level noise (MPEG or optic of the camera artefacts) and noise that has a semantic meaning (for example when someone passes just in front of the camera). In order to perform a reasonable detection of the transitions, we use the following assumptions:

- the lifetime in number of frames of a transition is included in a range between 1 and 20 frames
- the visual content of the video before and after a transition separating two different shots is usually significantly different

4.2.2 Statistical detection framework

We consider the frames content similarity before and after the boundaries of the homogeneous segments. If k is the index of the frame that separates the segments i and $i+1$, we use different windows of comparisons $k-l$ and $k+l$ for $l=1,2,4,8$. This informs us as to whether the transition is abrupt or gradual and gives an estimate of the length of the transition.

We measure the similarity in a very discriminative way and we take into account differences in color and spatial distribution of pixels information. We define the distances $D_{seg}(l)$ between the frames $k-l$ and $k+l$ as a vector containing the Jeffrey divergences of the color block histograms in the YUV color space and four rectangular blocks.

According to statistical detection theory, we formulate the detection problem as a binary hypothesis test by introducing:

- Hypothesis M: there is a transition between segments i and $i+1$
- Hypothesis $\overline{M}$: there is no transition present between segments i and $i+1$

The decision will be taken according to the minimum risk of error:

$$\frac{p(z|M)}{p(z|\overline{M})} \underset{M}{\overset{\overline{M}}{\lessgtr}} \frac{1-P_{k,l}(M)}{P_{k,l}(M)} \tag{14}$$

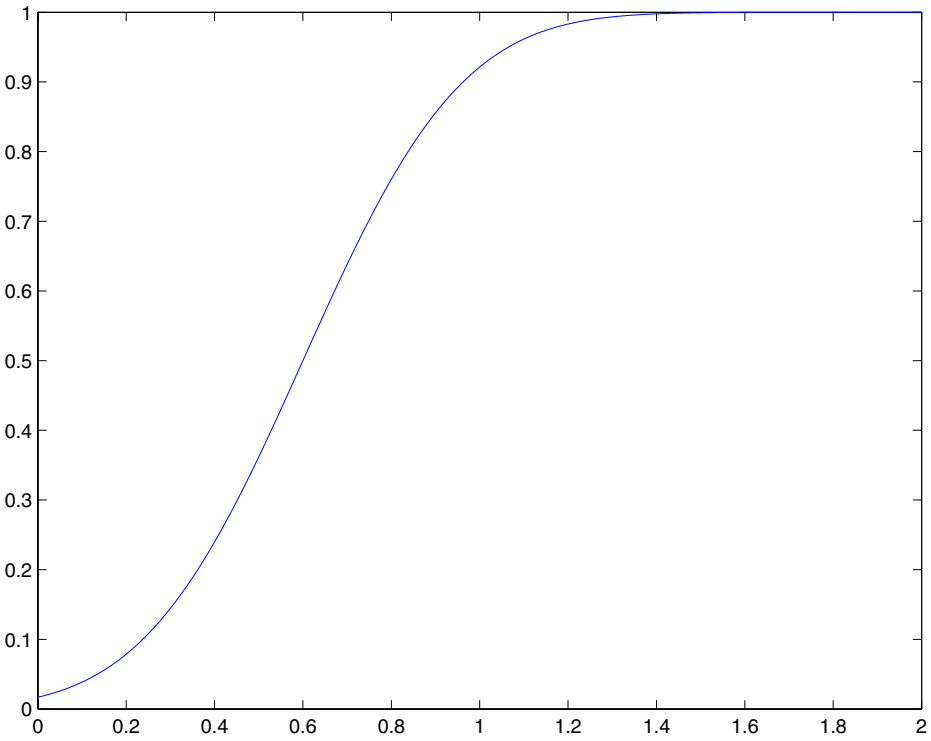

Fig. 2 Plot of the function used to design the conditional probability $P_{k,l}(M|D_{seg}(l))$

Using training data, we plot the shape of the distribution of the values of the distances when a transition is present and the likelihood function $p(z|M)$ can be found to be correctly approximated in the family of Gaussian functions:

$$p(z|M) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}} \tag{15}$$

Using the same method with the distribution of the values when no transition is present, the likelihood function $p(z|\overline{M})$ is found to belong to the family of Exponential functions:

$$p(z|\overline{M}) = e^{-hz} \tag{16}$$

Fig. 3 Keyframes extracted from the 'ariel' sequence where the transitions are very smooth dissolve effects

Table 1 Performances of the shot boundaries detection algorithm

Performances	Our algorithm	Hardcut detection alone
Recall	86.2	67.6
Precision	77.2	78.8

$P_{k,l}(M)$ is then defined as a conditional probability $P_{k,l}(M|D_{seg}(l))$ that depends of the dissimilarity of the frames before and after the transition.

$$P_{k,l}(M) = P_{k,l}(M|D_{seg}(l)) \tag{17}$$

The conditional probability $P_{k,l}(M|D_{seg}(l))$ (see figure 2) is computed using the similarity measure before and after the boundaries of the segments. The conditional probability should not be too sensitive when $D_{seg}(l)$ has extreme values. Between these values, the transition should be smooth (figure 3) to avoid the rejection of good candidates and this is the reason why $P_{k,l}(M|D_{seg}(l))$ can be chosen as:

$$P_{k,l}(M|D_{seg}(l)) = \frac{1}{2}\left(1 + erf\left(\frac{D_{seg}(l) - d_c}{\sigma_c}\right)\right) \tag{18}$$

The estimation of an approximate value for the length of the transition is found by finding the minimum l such that the detection of a transition is positive.

Using training data (see next paragraph), the parameters h, μ, σ are estimated and d_c and σ_c are chosen in order to maximize performances.

4.2.3 Results

We have experimented this method with 70 videos of the TREC Video Retrieval Evaluation (TRECVID) 2003 corpus using the evaluation framework of [11]. We used 35 hours of news programs coming from CNN and ABC. The ground truth has been obtained using a collaborative effort of the TRECVID community [12]. Each video contains around 400 transitions of every kind. There are many types of special effects involved because the videos come from television. The results are given in percentage in Table 1. We chose to accept a tolerance of 12 frames for the accuracy of the location of the transitions.

A better recall means there are a fewer number of missed detections while the better precision means that there are a fewer number of false detections.

The comparison in the Table 1 shows the advantage of using the information-based segmentation over the simple "hardcut" detection which misses all the gradual transitions. It will be interesting to compare our results with the results of the active participants of the TRECVID 2003 Workshop as soon as they will be publicly available.

5 Conclusion

In this paper, we have described an offline temporal segmentation algorithm based on the minimization of an information-based criterion. Our framework offers the advantages to be global and parameter-free. We first abstract the video content by a color dissimilarity profile (figure 4) which is then divided into dynamically homogeneous segments. The minimum message length (MML) criterion efficiently constrains the maximum-likelihood estimation

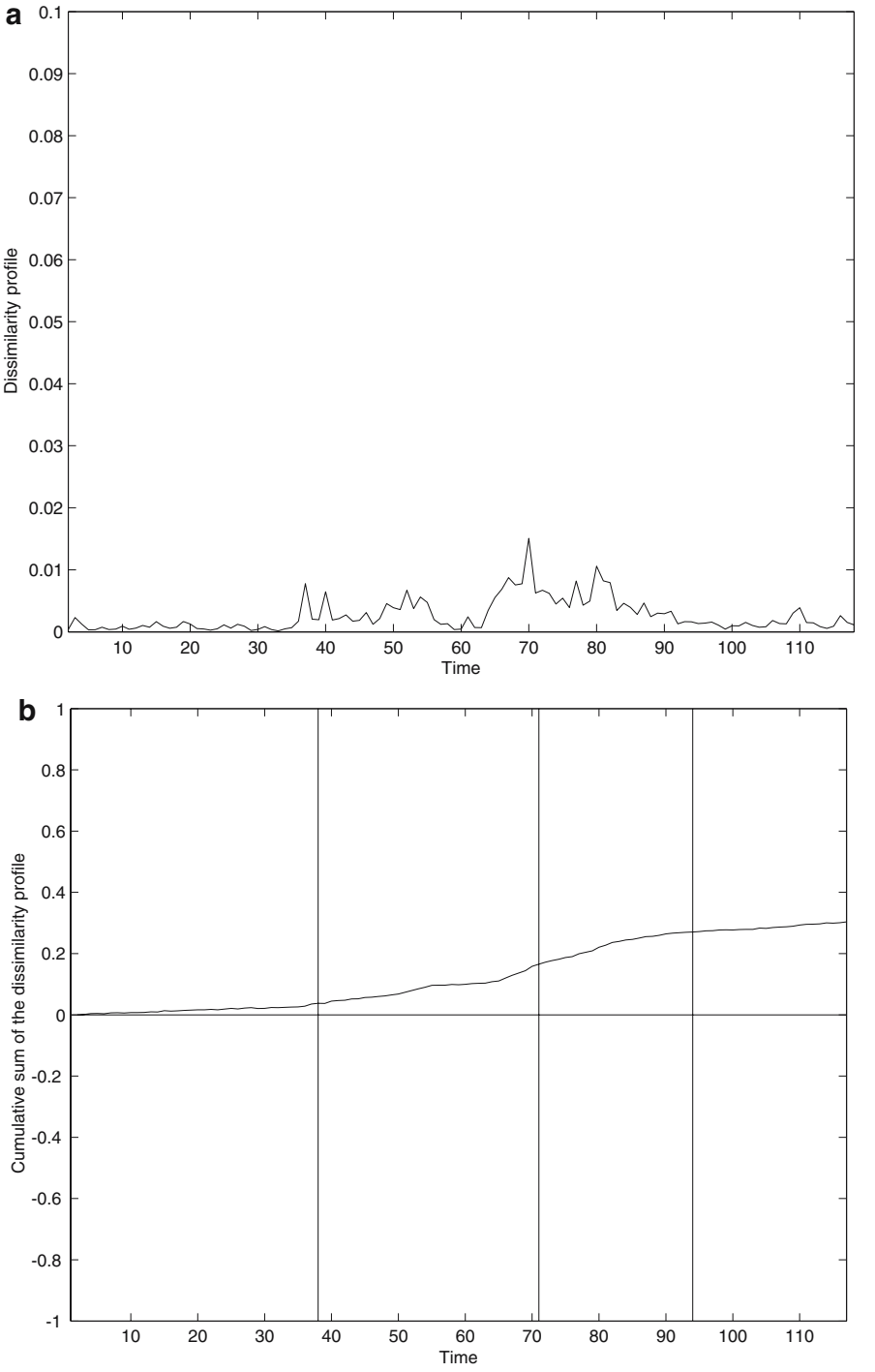

Fig. 4 (a) Dissimilarity profile of the 'ariel' sequence. (b) Trend of the dissimilarity profile and partitioning of the 'ariel' sequence

Fig. 5 Keyframe selection at different scales. The 60-min news report is summarized by (a) 394 keyframes (b) 197 keyframes (c) 98 keyframes

and offers the possibility to incorporate *a priori* knowledge. The minimization process is global and fast by using the characteristics of video data like the presence of hardcuts and redundancies. The segments can then be used as atomic components and reduce significantly the complexity of further analysis.

We then presented an original multiscale keyframe selection system (figure 5) that enables a user to interactively adjust the coarseness of the video representation. We have also shown that the detection of shot boundaries is highly simplified by disregarding the type of special effects involved during abrupt or gradual transitions and performs well using our methodology.

Future works includes the use of these atomic temporal segments for video representation and characterization. A better characterization of the video segments is, in our view, the only way to improve the accuracy of the results shown in Table 1. We will also be interested in analysing frequent patterns through a video database for summarization and categorization of video collections.

Acknowledgments This work is supported by the EU project M4–Multimodal Meeting Manager and the Swiss National Center of Competence IM2–Interactive Multimedia Information Management.

References

1. Gargi U, Kasturi R, Antani S (1998) Performance characterization and comparison of video indexing algorithms. Int. Conf. Computer Vision and Pattern Recognition (CVPR'98) 559–565.
2. Hanjalic A (2002) Shot-boundary detection: unraveled and resolved. IEEE Trans Circuits Syst Video Technol 12(2)
3. Heng WJ, Ngan KN (2002) Shot boundary refinement for long transition in digital video sequence. IEEE Trans Multimedia 4(4)
4. Koprinska I, Carrato S (2001) Temporal video segmentation: a survey. Signal Process, Image Commun 16(5):477–500
5. Lienhart R (1999) Comparison of automatic shot boundary detection algorithms. Storage and retrieval for still image and video databases, VII Proc. SPIE 3656–29
6. Fisher WD (1958) On grouping for maximum homogeneity. J Am Stat Assoc 53(284)
7. Fitzgibbon LJ, Allison L, Dowe DL (2000) Minimum message length grouping of ordered data. Proc. 11th International Conference on Algorithmic Learning Theory (ALT2000), 56–70
8. Baxter RA, Dowe DL (1994) Model selection in linear regression using the MML criterion. Proc. 4th IEEE Data Compression Conference
9. Wallace CS, Freeman PR (1987) Estimation and inference by compact coding. J R Stat Soc 49(3):240–265
10. Guigues L, Le Men H, Cocquerez JP (2003) Analyse et representation ensembles-echelle d'une image. 19e colloque du traitement du signal et des images (Gretsi'03), September
11. Ruiloba R, Joly P, Marchand-Maillet S, Quenot G (1999) Towards a standard protocol for the evaluation of video-to-shots segmentation algorithms. International Workshop in Content-Based Multimedia Indexing (CBMI)
12. Lin C-Y, Tseng BL, Smith JR (2003) Video collaborative annotation forum: establishing ground-truth labels on large multimedia datasets. Proceedings of the TRECVID 2003 Workshop, Gaithersburg, USA, November

Bruno Janvier received his MS degree from the Engineers School of Physics in Strasbourg, France in 2001, and is currently a PhD student with the *Viper* group of the Computer Vision and Multimedia Laboratory at University of Geneva, Switzerland. His research interest are temporal video segmentation, multimodal feature extraction, statistical learning of video content, and information retrieval.

Eric Bruno received his MS degree from the Engineers School of Physics in Strasbourg, France in 1995 and his PhD in signal processing from the Joseph Fourier University, Grenoble, France in 2001. Since 2002, he has been working at the Computer Vision and Multimedia Laboratory, University of Geneva, Switzerland as a research associate. His research interests focus on video analysis and content-based video indexing and retrieval (CBVR) which include motion estimation, region tracking, statistical learning of the video content and information retrieval.

Trierry Pun received a PhD degree in image processing from the Swiss Federal Institute of Technology, Lausanne, Switzerland in 1982. He joined the University of Geneva, Geneva, Switzerland in 1986, where he is currently a full professor of the Computer Science Department. Since 1979, he has been active in various domains of image processing, image analysis, and computer vision. He has authored or co-authored more than 140 journal and conference papers in these areas and led or participated in a number of national and European research projects. His current research interest is focused on several aspects of the design of multimedia information systems: image and video content-based information retrieval systems, Web browser for blind users, and image and video watermarking.

Stéphane Marchand-Maillet received his PhD in theoretical image processing from Imperial College, London in 1997. He then joined the Institut Eurecom at Sophia-Antipolis (France) where he worked on automatic video indexing techniques based on human face localization and recognition. Since 1999, he is assistant professor in the Computer Vision and Multimedia Lab at the University of Geneva, where he is working on content-based multimedia retrieval as head of the *Viper* research group. He has authored several publications on image analysis and information retrieval, including a book on low-level image analysis. He currently leads the Benchathlon, a joint international effort for benchmarking content-based image retrieval systems.