Fast and Flexible Video Content Browsing based on H.264/AVC Features

[Demonstration]

Klaus Schöffmann
Department of Information Technology (ITEC)
Klagenfurt University
9020 Klagenfurt, Austria
ks@itec.uni-klu.ac.at

Laszlo Böszörmenyi
Department of Information Technology (ITEC)
Klagenfurt University
9020 Klagenfurt, Austria
laszlo@itec.uni-klu.ac.at

ABSTRACT

We present a video content browsing tool which is supposed to be used for two purposes: (1) efficiently searching a certain scene within a long video sequence and (2) quickly identifying videos out of interest. Instead of a complex user interface with many features we rather use a simple but flexible player-like interface which can, however, support the user on taking advantage of his/her knowledge about the content semantics. The content analysis is based on some simple features which can be extracted from compressed data of H264/AVC and, thus, allow very fast analysis. An early user study has shown encouraging results.

1. INTRODUCTION

The process of rapidly viewing the content of a video sequence is denoted as video browsing[1]. Video browsing tools should support the user at the difficult task of finding a short scene within a (maybe long) video sequence. A remarkable amount of research has already been published for this topic (e.g. [1, 2, 3, 4, 5]). Usually they are based on some content analysis of the underlying video sequence. However, the problem of all those proposals is that they are based on content analysis of pixel data often not quickly available due to the fact that most videos are stored in compressed form. Moreover, proposed video browsing tools do often completely destroy the usual chronological interaction model users have learned at searching videos since the invention of VCRs in the 1960s.

Here we present a video browsing tool which uses a player-like navigation/interaction model and is based on some simple features that can be efficiently extracted from compressed data of a video. The tool allows both to use a player-like navigation within a video with some helpful extensions and exploiting a user's knowledge about the content semantics/characteristics. For instance, if a users knows that the searched content will appear in the last part of the video (e.g. the final scenes in a soccer match), the user can start playback for several areas in the last part in simultaneous manner in order to find quickly the wished content. As another example imagine a ski-jumping video where the user wants to see some competitors driving down the jump. In that case the user can benefit from his/her knowledge about the content semantics and quickly select those scenes (i.e. shots) of the videos containing a lot of *motion down*.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
NOSSDAV '08 Braunschweig, Germany
Copyright 2008 ACM 978-1-60588-157-6/05/2008 ...$5.00.

2. FLEXIBLE USER INTERFACE

As shown in Figure 2 our interface uses parallel video windows whereas the user can choose if more or less windows should be displayed at a glance. In each window a key-frame of the corresponding content is shown as a preview. A user can start video playback for a window by simply clicking on it. It is also possible to start playback for more than one window. Moreover, a user may change the criteria and the order of sorting of the windows, switch one window to fullscreen mode or use the corresponding time-slider to navigate within one window. We provide three different navigation modes:

(1) In the **chronological mode** shots of the video are shown in chronological order. If the video contains more shots than currently displayed windows, a paging mechanism can be used to quickly step through all available shots. For shot boundary detection we use an efficient method[6] working in the compressed domain of H.264/AVC.

(2) In the **feature-based mode** shots are sorted according to a priority value based on a certain low-level feature selected by the user (e.g. *homogeneity of content* in the shot, average *brightness* of shot, average amount of *motion going to a specific direction* in the shot, etc.). In our current prototype the features are extracted from compressed domain of H.264/AVC.

(3) In the **hierarchical mode** the content of the video is uniformly divided over all video windows in one page. For instance, if the user selected four windows, the window left above represents the first quarter of the entire video content, the window right above represents the second quarter and so on. A user may go down the hierarchy and further divide the content of one window by simply right-clicking the video window. No content analysis is used for this navigation mode.

Let us assume a user is browsing a recording of a ski-jumping event where he/she wants to see the final results.

Based on the user's knowledge that results are usually shown at the end of the competition, the user can quickly find the results by activating chronological mode in reverse order. This example is shown in Figure 1 where the searched scene appears in the second video window.

Figure 1: Chronological View

However, as already mentioned earlier, instead of the results a user might be rather interested in scenes showing competitors driving down the jump. In that case the user may probably know that such scenes contain a lot of motion down. Thus, he/she may switch to feature-based mode and select the average amount of motion going down within a shot as a feature. As illustrated in Figure 2, the searched scenes will quickly appear at the beginning of the shot list.

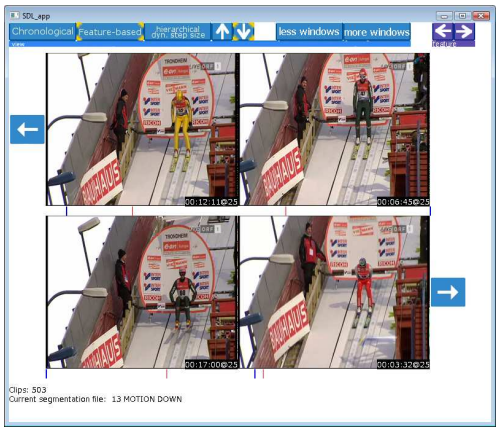

Figure 2: Feature-based View

As another example, consider the case a user is browsing a recording of daily news where the weather forecast should be found. As the user knows the weather forecast is typically at the end of the news he/she selects hierarchical mode to uniformly divide the content into e.g. nine areas (and maybe start playback for all those nine windows). As you can see in Figure 3, the searched content appears in the last area at the top level of the hierarchy[1]. However, the user may further divide that area by simply clicking on it which will give a more refined view of the weather forecast in the second level of the hierarchy.

[1]The grey bars at the bottom of each window denote the position of the content in relation to the entire video.

Figure 3: Hierarchical View

3. CONCLUSIONS AND FUTURE WORK

The presented video content browsing tool provides a high benefit to users who have some knowledge about the content they are searching in. Moreover, it allows a user to efficiently identify content out of interest due to the possibility of parallel playback. The content processing is based on some simple features which can be extracted from the compressed domain and, thus, allow very fast analysis. Early user studies have shown that users accept the interface and can highly benefit at searching if they have some knowledge about the content. In future work we will evaluate more features for feature-based mode, extend the tool to support video archives rather than single video sequences, and further improve the user interface.

4. REFERENCES

[1] F. Arman, R. Depommier, A. Hsu, and M.-Y. Chiu. Content-based browsing of video sequences. In *MULTIMEDIA '94: Proceedings of the second ACM international conference on Multimedia*, pages 97–103, NY, USA, 1994. ACM.

[2] W. Bailer, C. Schober, and G. Thallinger. Video content browsing based on iterative feature clustering for rushes exploitation. *Proceedings of TRECVID Workshop*, pages 230–239, 2006.

[3] H. Eidenberger. A Video Browsing Application based on visual MPEG-7 Descriptors and Self-Organising Maps. *International Journal of Fuzzy Systems*, 6(3):124–137, 2004.

[4] W. Hurst and P. Jarvers. Interactive, Dynamic Video Browsing with the Zoomslider Interface. *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 558–561, 2005.

[5] N. Moraveji. Improving video browsing with an eye-tracking evaluation of feature-based color bars. *Digital Libraries, 2004. Proceedings of the 2004 Joint ACM/IEEE Conference on*, pages 49–50, 2004.

[6] K. Schöffmann and L. Böszörmenyi. Fast Segmentation of H.264/AVC Bitstreams for On-Demand Video Summarization. In S. Satoh, F. Nack, and M. Etoh, editors, *Advances in Multimedia Modeling*, volume 4903 of *LNCS*, pages 265–276. Springer, 2008.