Video browsing interfaces and applications: a review

Klaus Schoeffmann,[a] Frank Hopfgartner,[b] Oge Marques,[c]
Laszlo Boeszoermenyi,[a] Joemon M. Jose[b]
[a] University of Klagenfurt, Universitaetsstrasse 65-67, 9020 Klagenfurt, Austria
[b] University of Glasgow, Department of Computing Science, 18 Lilybank Gardens,
Glasgow G12 8RZ, United Kingdom
[c] Florida Atlantic University, 777 Glades Road, Boca Raton, Florida 33431-0991, USA

Abstract. We present a comprehensive review of the state of the art in video browsing and retrieval systems, with special emphasis on interfaces and applications. There has been a significant increase in activity (e.g., storage, retrieval, and sharing) employing video data in the past decade, both for personal and professional use. The ever-growing amount of video content available for human consumption and the inherent characteristics of video data—which, if presented in its raw format, is rather unwieldy and costly—have become driving forces for the development of more effective solutions to present video contents and allow rich user interaction. As a result, there are many contemporary research efforts toward developing better video browsing solutions, which we summarize. We review more than 40 different video browsing and retrieval interfaces and classify them into three groups: applications that use video-player-like interaction, video retrieval applications, and browsing solutions based on video surrogates. For each category, we present a summary of existing work, highlight the technical aspects of each solution, and compare them against each other. *© 2010 Society of Photo-Optical Instrumentation Engineers*. [DOI: 10.1117/6.0000005]

Keywords: video browsing; video navigation.

Paper SR090108 received Nov. 4, 2009; accepted for publication Dec. 10, 2009; published online Mar. 16, 2010.

1 Introduction

The main research motivation in interactive information retrieval is to support users in their information-seeking process. Salton [1] defines a classical information-seeking model as follows. Triggered by an information need, users start formulating a search query, inspect retrieval results, and, if needed, reformulate the query until they are satisfied with the retrieval result. Belkin et al. [2] extend this model further by distinguishing between querying/searching for results, usually by triggering a new search query, and browsing/navigating through the already retrieved results. However, users of information retrieval systems have very often only a very fuzzy understanding of how to find the information they are looking for. According to Spink et al. [3], users are often uncertain of their information need and hence have problems finding a starting point for their information-seeking task. And even if users know exactly what they are intending to retrieve, formulating a "good" search query can be a challenging task. This problem is exacerbated when dealing with multimedia data. The formulation of a search query hence plays an important role in this task. Graphical user interfaces serve here as a mediator between the available data corpus and the user. It is the retrieval systems' interface that will provide users facilities to formulate search queries and/or to dig into the available data. Hearst [4] outlines various conditions that dominate the design of state-of-the-art search interfaces. First of all, the process of searching is a means toward satisfying an information need. Interfaces should therefore avoid being intrusive, since this could disturb the users in their seeking process. Moreover, satisfying an information need is already a mentally intensive task. Consequently, the interface should not distract the users, but rather support them in their assessment of the search results. With the advent of the World Wide Web, search interfaces are used not only by high-expertise librarians but also by the general public. Therefore, user interfaces have to be intuitive to use by a diverse group of potential users. Consequently, widely used web search interfaces such as Google, Bing, or Yahoo! have very simple interfaces, mainly consisting of a keyword search box and results that are displayed in a vertical list.

Considering the success of the above-mentioned web search engines, it is not premature to assume that these interfaces effectively handle the interaction between the user and the underlying text retrieval engine. However, text search engines are rather simple in comparison to their counterparts in the video retrieval domain. Therefore, Jaimes et al. [5] argue that this additional complexity introduces further challenges in the design of video retrieval interfaces.

The first challenge is how users shall be assisted in formulating a search query. Snoek et al. [6] identified three query formulation paradigms in the video retrieval domain: query by textual keyword, query by visual example, and query by concept. Query by textual keyword has been largely studied in the last decades and thus is a well-established search paradigm. Visual queries arise from content-based image retrieval systems. Users can provide an example image, select a set of colors from a color palette, or sketch images and the underlying retrieval engine retrieves visually similar images. Query by concept includes the allocation of low-level features to high-level concepts. Basic examples are concepts such as outdoor vs indoor [7] and cityscape vs landscape [8], which can be identified based on visual features. Concepts can be used to filter search results, e.g., by displaying only the results that depict a landscape. Video retrieval interfaces need to be provided with corresponding query formulation possibilities in order to support these paradigms. Another challenge is how videos shall be visualized to allow the user an easy understanding of the content. In the text retrieval domain, short summaries, referred to as snippets, are usually displayed, which allow the users of the system to judge the content of the retrieved document. Much of the research (e.g., Refs. 9 and 10) indicates that such snippets are most informative when they show the search terms in their corresponding context. Considering the different nature of video documents and query options, identifying representative video snippets is a challenging research problem. Moreover, another challenge is how users can be assisted in browsing the retrieved video documents. Systems are required that enable users to interactively explore the content of a video in order to get knowledge about its content.

In this paper, we survey representative state-of-the-art video browsing and exploration interfaces. While research on video browsing was already very active in the 1990s (e.g., see Refs. 11–26), in this paper we focus on video browsing approaches that have been presented in the literature during the last 10 years. Many systems reviewed in this paper have been evaluated within TRECVID [27], a series of benchmarking workshops aimed at improving content-based video retrieval techniques. The paper is structured as follows. In Section 2, we review video browsing applications that rely on interaction similar to classical video players. Section 3 introduces applications that allow users to explore the video corpus using visual key frames. Section 4 surveys video browsing applications that visualize video content in unconventional ways. The paper concludes in Section 5.

2 Video Browsing Applications Using Video-Player-Like Interaction

Common video players use simple interaction as a means to navigate through the content of a video. However, although these interaction methods are often employed for the task of searching, they are mostly unsatisfying. Therefore, many efforts have been made to extend the simple video-player interaction model with a more powerful means for content-based search. In this chapter we review such video browsing applications, which can be characterized as "extended video players."

One of the early efforts in this direction was done by Li et al. [28] in 2000. They developed two different versions of a video browser, a basic browser and an enhanced browser, and compared both versions in a user study with 30 participants. The basic browser included basic controls that are typically provided by video players, such as play, pause, fast-forward, seeker bar, etc. The enhanced browser provided several additional features:

- The time compression (TC) function increases/decreases playback speed from 50% to 250% while always preserving the audio pitch.
- The pause removal function removes segments that seem to contain silence or a pause, according to the audio channel.
- The table of contents (TOC) feature is a list of textual entries (e.g., for "classroom" videos, generated from the corresponding slides).
- A visual index contains key frames of all the shots.
- The jump feature allows the user to jump backward or forward by 5 or 10 sec, jump to the next note, or jump to the next slide transition ("classroom" videos) or shot change (shot boundary seek).

In their evaluation, they showed that users of the enhanced browser rated TC and TOC as the most useful features while the shot seek feature was used most often. Moreover, their evaluation showed that participants spent considerably less time watching videos with the default playback speed when using the enhanced browser. It also revealed that the fast-forward feature of the basic browser was used significantly less than the seeker bar. For the classroom and the news video, fast-forward was almost never used. However, for the sports category (baseball video) the average number of fast-forward usage heavily increased for both the basic and the enhanced browser because it allowed higher speed-up than TC. Participants agreed that especially in the sports and news categories, having enhanced browsing features would be of great benefit and affect the way they watch television.

Barbieri et al. [29] presented the concept of the color browser, where the background of a typical seeker bar is enhanced by vertical color lines, representing information about the content. As information to be presented in the vertical lines they used (1) the dominant color of each corresponding frame (Fig. 1) and (2) the volume of the audio track. For the dominant colors a smoothening filter is applied to filter out successive heavily changing color values (see Fig. 1). As there is not enough space to display a vertical line for every frame in the background of a seeker bar, they proposed to use two seeker bars. The first one acts as a fast navigation means with different time scales for every video sequence and the second acts as a time-related zoom using the same time scale for every video sequence. They argued that the fixed time scale of the zoomed seeker bar would enable a user to "learn to recognize patterns of colors within different programs."

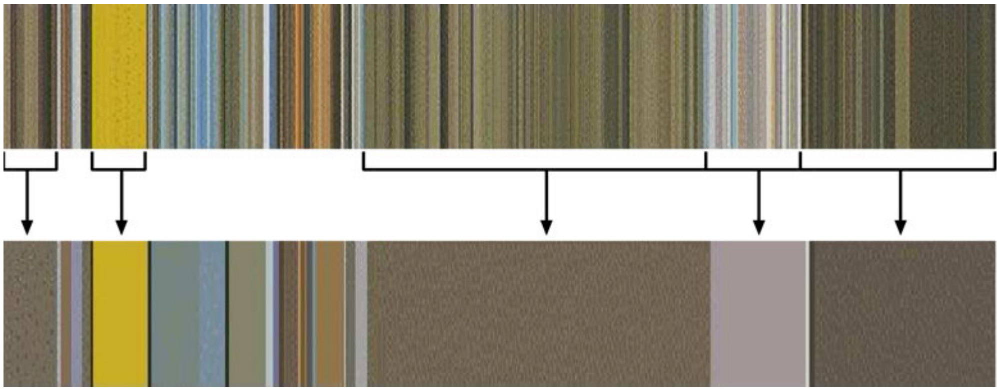


Fig. 1 Visualization of the ColorBrowser without (above) and with (below) a smoothening filter. [29] © 2001 IEEE.

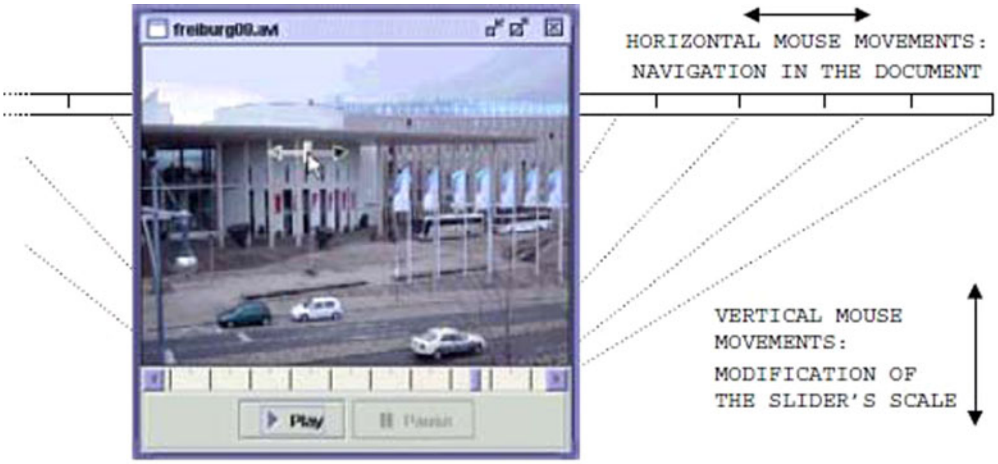


Fig. 2 Video navigation with the Zoom Slider. [36] © 2005 IEEE.

Tang et al. [30] presented the NewsEye, an application for improved news story browsing. With unsupervised fuzzy c-means clustering the content is first segmented into shots. Then, the shots are grouped together in order to form several different news stories. For that purpose they use a graph-theoretical cluster analysis algorithm to identify all shots that show an anchorperson. Furthermore, they also use optical character recognition (OCR) to detect caption text in the frames of a news story. Their video-player-like interface contains a panel showing key frames of all the shots in the current news story as well as the detected caption text. Their application also provides a keyword-based search function for the caption text.

Divakaran et al. [31] proposed a video summarization method that can also be used for video browsing. Their approach takes advantage of information extracted from the compressed domain of the video and it is based on the hypothesis that the intensity of motion activity is a measure of the summarizability. To skip over parts of the video with low motion activity, the playback rate is adjusted dynamically. They also analyze the audio channel in order to detect speaker changes and to generate a list of included topics.

In a further work, Peker and Divakaran [32] propose the use of adaptive fast playback (AFP) for the purpose of quickly skimming through a video sequence. The AFP approach is used accordingly to the level of complexity of a particular scene and the capabilities of the human visual system. The level of complexity is determined based on the amount of motion and spatial-temporal complexity of a scene. Thus, scenes with low complexity are played faster while scenes with high complexity are played at a lower speed.

Liu et al. [33] presented a news video browsing system called NewsBR, which is very similar to the NewsEye system [30]. It performs story segmentation and caption text extraction. The story segmentation uses a shot detection method based on χ^2 histogram matching and silence clip detection. For caption text extraction they classify frames into topic-caption frames and non-topic-caption frames. To topic-caption frames, which are those that contain text of a news topic, a horizontal and vertical Sobel filter is applied before an OCR library is used to detect the text. Their interface shows a TOC (in combination with a key frame preview) according to the story segmentation, which can be used as a means of navigation. It also provides a keyword-based search on the extracted caption text.

Moraveji [34] proposed the assignment of unique and visually distinctive colors to particular content features, such as persons, faces, vehicles, etc. These colors can be further used for visualization in a timeline that shows "the most relevant" feature for a particular segment of frames. When the mouse is moved over a particular segment, some additional information—such as the concept/feature represented by the color—is displayed below. A click on a color bar in the timeline will start video playback from the corresponding time position. The work of

Moraveji is similar to the work of Barbieri et al. [29], as it is based on the idea of enhancing the timeline (or background of a seeker bar) with content information.

To overcome the limitations of typical seeker bars in standard video players, Hürst et al. proposed the ZoomSlider interface [35,36]. Instead of a common seeker bar the entire player window is used as a hidden seeker bar with different stages of granularity in a linear way (see Fig. 2). When the user clicks on any position in the player window, a seeker bar for moving backward or forward appears. The granularity of that seeker bar is dependent on the vertical position of the mouse in relation to the entire height of the player window. When the mouse is moved in a vertical direction, the scaling of the seeker bar changes in a linear way. The finest granularity is used at the top of the window and the coarsest granularity is used at the bottom of the window. Therefore, a user can zoom-in or zoom-out the scaling of the seeker bar by selecting different vertical mouse positions.

The concept of the ZoomSlider interface has been extended in Ref. 37 to additionally provide similar mechanisms for changing the playback speed of the video. The right vertical part of the player window is used to change the playback speed where the slowest speed is assigned to the top and the highest speed is assigned to the bottom of the player window. The user can select any playback speed in a linear fashion based on the vertical mouse position. The same manner is used for backward playback at the left vertical part of the window. In Refs. 38 and 39 the idea has been further adapted for mobile devices, where the entire screen is used for video playback containing "virtual" seeker bars in the same way.

Divakaran and Otsuka [40] argued that "Current personal video recorders can store hundreds of hours of content and the future promises even greater storage capacity. Manual navigation through such large volumes of content would be tedious if not infeasible." Therefore, they presented a content-based feature visualization concept (Fig. 3), which is based on classification of audio segments into several different categories (e.g., speech, applause, cheering, etc.). An importance level is calculated according to these categories and plotted in a two-dimensional

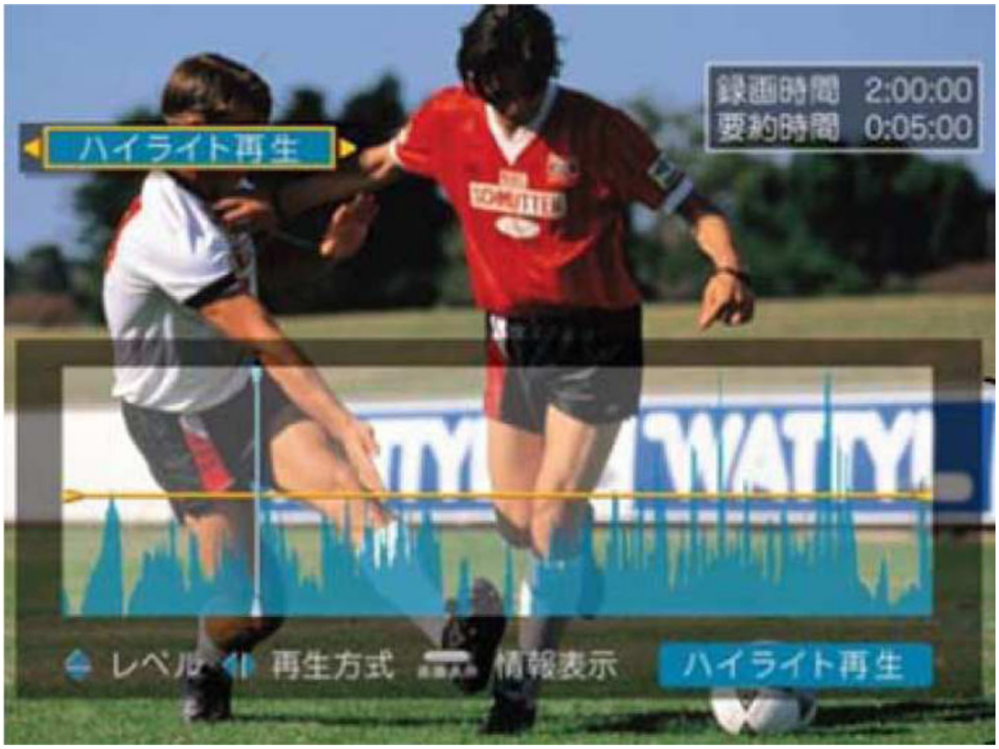


Fig. 3 A video browsing enhanced personal video recorder. [40] © 2007 IEEE.

graph, which can be shown as a timeline overlay onto the original content. The user can set an importance level threshold (yellow line in the figure), which is used by the system to filter out all the content having a lower importance level. In other words, a “highlight search” function is available to the user. They evaluated their concept with several sports videos in a user study, which showed that users like the importance level plot due to its flexibility, even if the visualization results in mistakes. The concept has been integrated into a Personal Video Recorder product sold by Mitsubishi Electric in Japan.

An interesting approach for video browsing by direct manipulation was presented by Dragicevic et al. [41] in 2008. As a complement to the seeker bar they propose relative flow dragging, which is a technique to move forward and backward in a video by direct mouse manipulation (i.e., dragging) of content objects. They use an optical flow estimation algorithm based on scale-invariant feature transform (SIFT) [42] salient feature points of two consecutive frames. A user study has been conducted and it has shown that relative flow dragging can significantly outperform the seeker bar on specific search tasks.

A system very similar to that of Dragicevic et al. was already proposed by Kimber et al. in 2007 [43]. In similarity, their system shows motion trails of objects in a scene, based on foreground/background segmentation and object tracking, and allows an object to be dragged along a trail with the mouse. For an application in a floor surveillance video they additionally show the corresponding floor plan including motion trails.

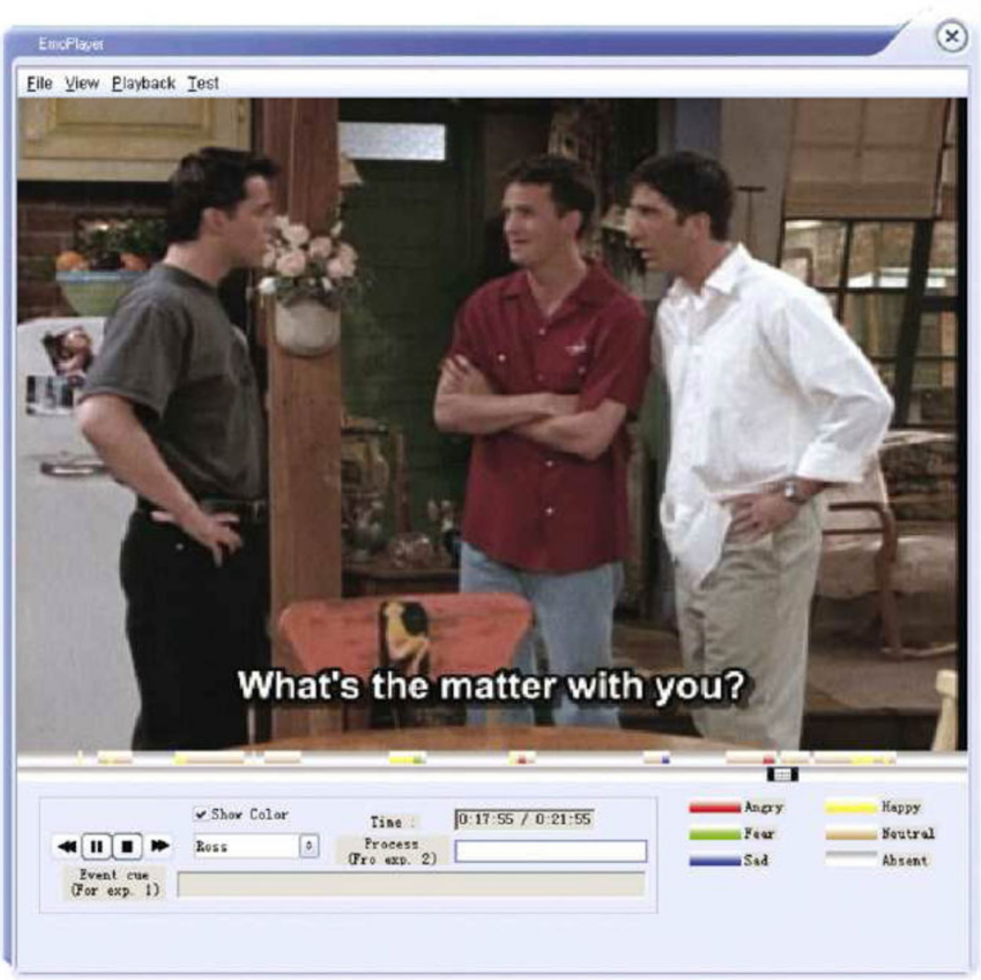
Chen et al. presented the EmoPlayer [44], a video player that can visualize affective annotations. In particular, different emotions of actors and actresses—angry, fear, sad, happy, and neutral—can be visualized for a selected character in a video, based on a manually annotated XML file. The emotions are shown in a color-coded bar directly above the usual seeker bar. Different colors are used for different emotions (see Fig. 4). If a character is not present in a specific scene the bar shows no color (i.e., white) for the corresponding segment. Therefore, a user can simply identify in which segments a particular character is present and which emotion the character expresses.

In 2008, Yang et al. [45] proposed the smart video player to facilitate browsing and seeking in videos. It provides a filmstrip view in the bottom part of the screen, which shows key frames of the shots of the video (see Fig. 5). The user can set the level of detail for that view and, thus, extend or reduce the number of shots displayed within the filmstrip. The Smart Video Player does also contain a recommendation function to present a list of other similar videos to the user.

In 2002, a similar technique was presented by Drucker et al. [46] with the SmartSkip interface for consumer devices (e.g., VCRs). They propose a thumbnail view at the bottom of the screen that can be used to skip over less-interesting parts of the video. These thumbnails have been uniformly selected from the content of the video although they experimented with a shot-based view as well. The shot-based view, however, was omitted after user tests. The reason was that for communicating the actual time between shots the spatial layout was changed to a nonuniform manner, which users disliked. The level of detail of the thumbnail view can be configured by users ranging from 10 sec all the way up to 8 min.

Hiu and Zhang [47] combine video event analysis with textual indices for the SportBR video browser, which can be used for browsing soccer videos. In particular, they use the color layout of example images of penalty kicks, free kicks, and corner kicks and search for similar scenes in the video. In order to improve the accuracy of the event detection, speech analysis (detection of some specific words) is performed. Moreover, they use an OCR algorithm to detect text that appears in the frames within detected events. The interface of their application allows (1) improved navigation within the video based on the detected events and (2) keyword search based on speech and text detection.

Vakkalanka et al. [48] presented the NVIBRS, a news video indexing, browsing, and retrieval system. Their system performs shot detection and news story segmentation based on localization of anchorperson frames. To detect anchorperson frames they first classify all frames into high motion and low motion. On low-motion frames they apply a face detection method based on a



☑ Show Color

Ross

Time : 0:17:55 / 0:21:55

Fig. 4 Video browsing with the EmoPlayer. [44]

Gaussian mixture model for skin-color detection. When a face has been detected the location of the eyes and the mouth is estimated and features are extracted from those regions.

The feature vectors are used as input for an anchorperson classifier working with autoassociative neural network models. The interface of their application provides a tree view of all detected news story units in a video and shows key frames of the currently selected story as a navigation means. It also allows a user to perform a textual query by specifying the desired video category as the news content is categorized into a few categories.

Rehatschek et al. [49] and Bailer et al. [50] presented the semantic video annotation tool (SVAT), a tool that is basically intended to be used for video annotation (see Fig. 6*). However, in order to improve navigation within a single video for faster annotation they developed several advanced navigation functions. In particular, they provide a video-player-like component (1) in combination with a temporal visualization of shot boundaries, key frames, stripe images, and motion events (pan, zoom, etc.) as a means of navigation (2). Their interface also includes a shot list (3), a list for selected key frames (4), and an annotation view (5) to add textual information to shots and key frames. Moreover, the tool includes a SIFT-based automatic similarity search

*Screenshot of a trial version that has been downloaded from ftp://iis.joanneum.at/demonstrator.

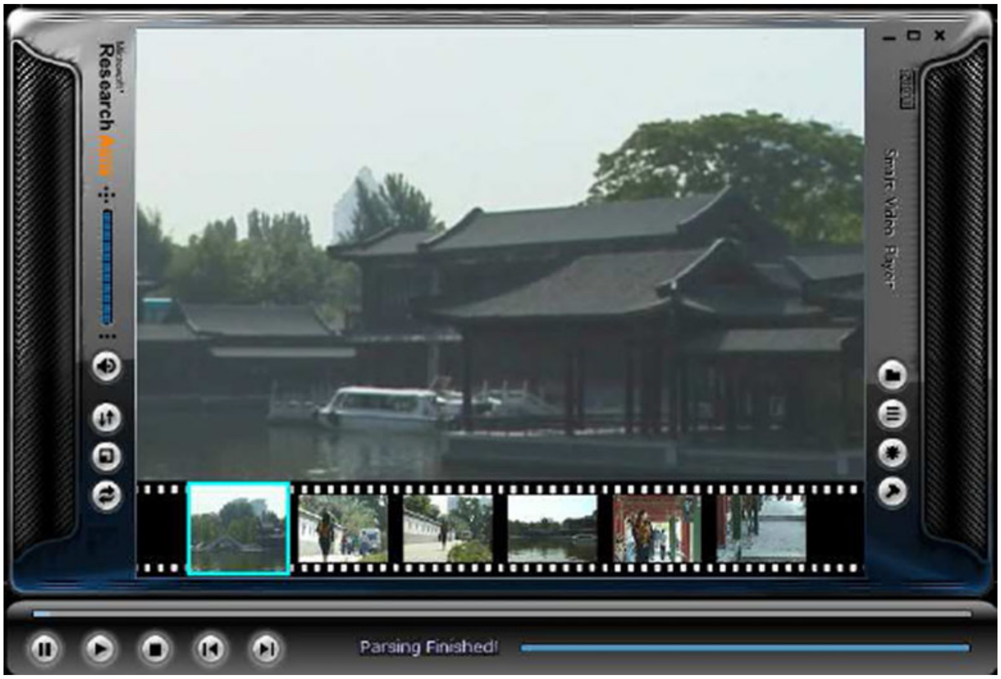


Fig. 5 The smart video player. [45] © 2008 IEEE.

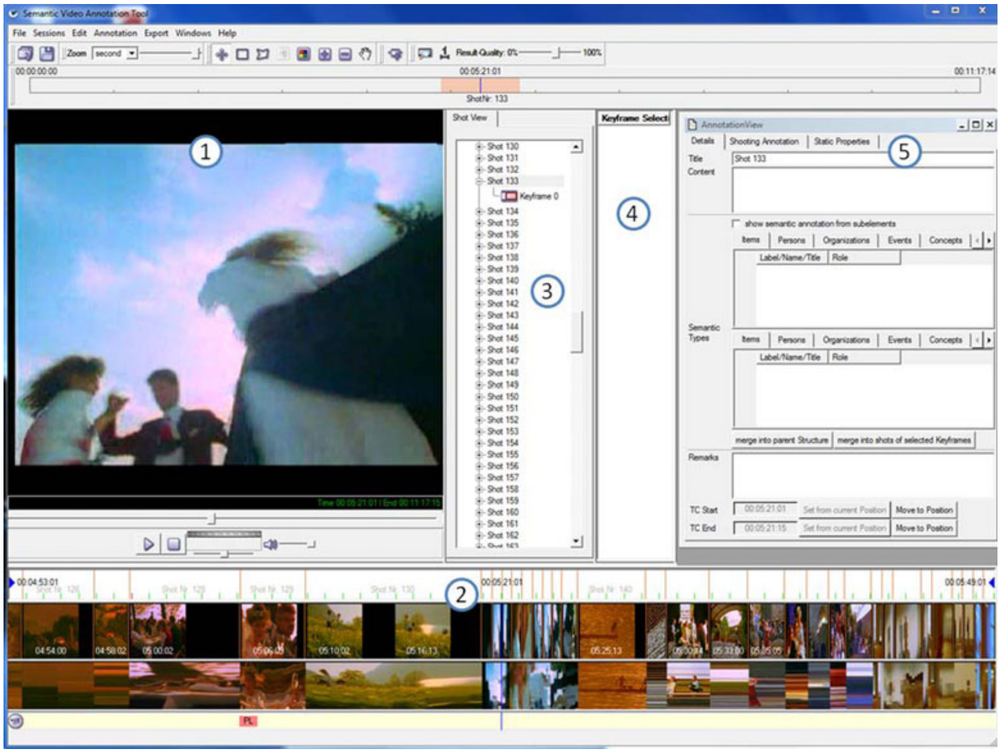


Fig. 6 The semantic video annotation tool (SVAT). [49,50]

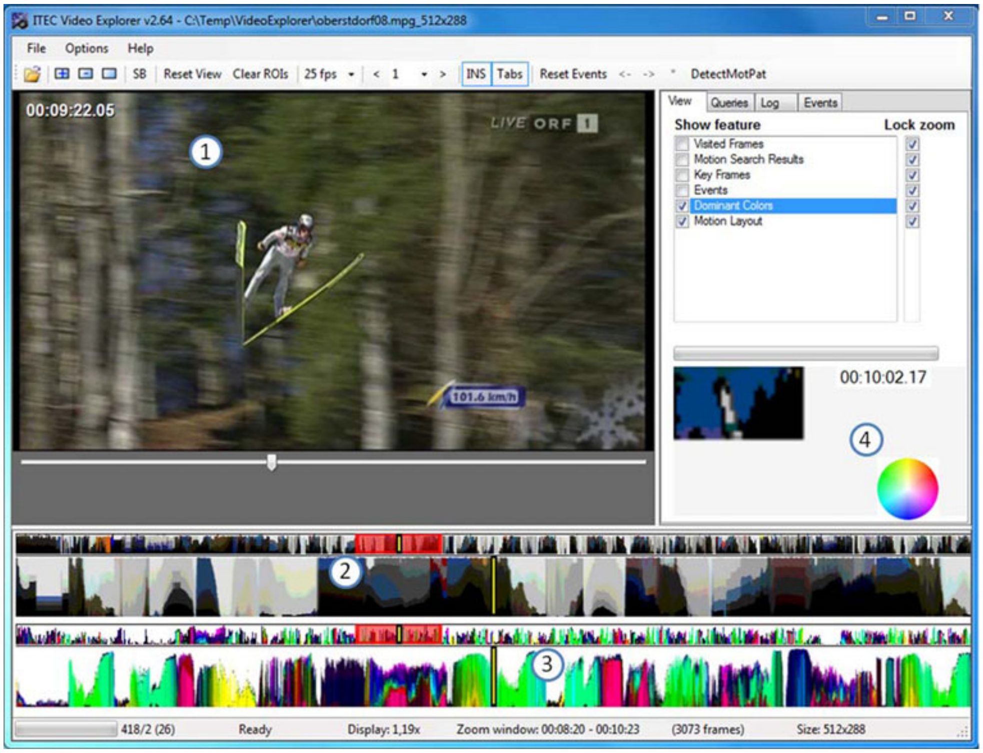


Fig. 7 Video browsing with the video explorer. [53]

function that can be used to find similar content in the video according to a user-defined region of interest.

A similar tool for the explicit purpose of video browsing has been presented by Schoeffmann et al. [51]. Their video explorer uses the concept of interactive navigation summaries (INSs) in order to help a user with the task of navigation through a video. INSs can effectively visualize several time-related pieces of information. As shown in Fig. 7, the video explorer consists of a video-player-like component (1) and a few INSs (2 and 3) that act as an alternative to the common seeker bar. In Fig. 7, (2) shows the dominant color INS and (3) shows the motion layout INS. While the dominant color INS [52] visualizes the temporal flow of the dominant colors, the motion layout INS [53] visualizes the temporal flow of motion characteristics. More precisely, for the second INS, motion vectors of H.264/AVC compressed video files are extracted, classified by direction and intensity, and visualized in an HSV color representation. A hue circle of the HSV color space is shown at (4) in order to give the user a hint as to which color is used to visualize a particular direction (e.g., blue for downward motion, yellow for upward motion, red for motion that is upward to the right, and so on). The visualization shows both how much motion in a specific direction every frame contains [the amount of a specific color (H) in a vertical line] and how fast this motion is [intensity (V) of the color]. For a specific scene this yields to a certain motion pattern that can help users to interactively detect similar scenes in the video, as they appear with similar motion patterns in the visualization. Figure 7 shows an example of a ski-jumping video where jump-offs of competitors are visualized as greenish V-like patterns. In order to preserve the browsing context their model of an INS contains an overview visualization, including a zoom window, and a detailed visualization. While the overview visualization represents the entire video in low quality, the detailed visualization (located directly below to the overview) shows all the details of a particular segment. The zoom window (shown as a red box) determines the position and duration of this segment to be shown in the detailed visualization of the corresponding INS.

Cheng et al. [54] proposed the SmartPlayer for browsing the content of a video. In addition to manually changing the playback speed, it provides an automatic playback speed adaptation according to scene complexity, which is computed through motion complexity analysis for every shot. The player has been designed in accordance with the "scenic car driving" metaphor, where a driver slows down at interesting areas and speeds up through unexciting areas. The SmartPlayer also learns the users' preferences of playback speed for specific type of video content.

Tables 1 and 2 give an overview of approaches reviewed in this section. All applications in this section and in the two subsequent ones have been structured by the following criteria:

- Is there support for browsing and querying the content?
- Is the application intended to be used with a single video file (1) or an archive (N)?
- What is the smallest structuring unit the content analysis and interaction is bound to?
- What is the video content domain the application is designed for?
- Which content analysis is used?
- How is the content visualization/representation and user interaction implemented?

3 Video Browsing Concepts in Video Retrieval Applications

While browsing videos using a video-player-like interaction scheme is useful in some scenarios, this approach cannot easily be adopted in interactive video retrieval. In contrast to video browsing, where users often just interactively browse through video files in order to explore their content, a video retrieval user wants to search certain scenes in a collection of videos. Such a user is typically expected to know quite exactly what he or she is looking for. Therefore, it is crucial to provide appropriate search functions for different types of queries. However, at least for the task of presenting the results to a query, a video retrieval application needs to consider video browsing concepts as well. Furthermore, video browsing mechanisms are often combined with video retrieval methods (e.g., in VAST MM [55]) in order to serve all different types of users. Nowadays interactive web-based video retrieval is also getting more important as both retrieval giants Yahoo! and Google are working on their own video retrieval engines. In addition, there are numerous video search engines such as www.truveo.com and www.blinkx.com that offer similar services. These online video platforms allow users to upload and share their own videos. The data set of such platforms grows extremely quickly and necessitates new ways for allowing users to efficiently browse through a large collection of videos. Since a review on arising challenges in the multimedia retrieval domain is out of the scope of this paper, the interested reader is referred to Veltkamp et al. [56] for further reading. In this section we focus on different interface designs used in video retrieval systems.

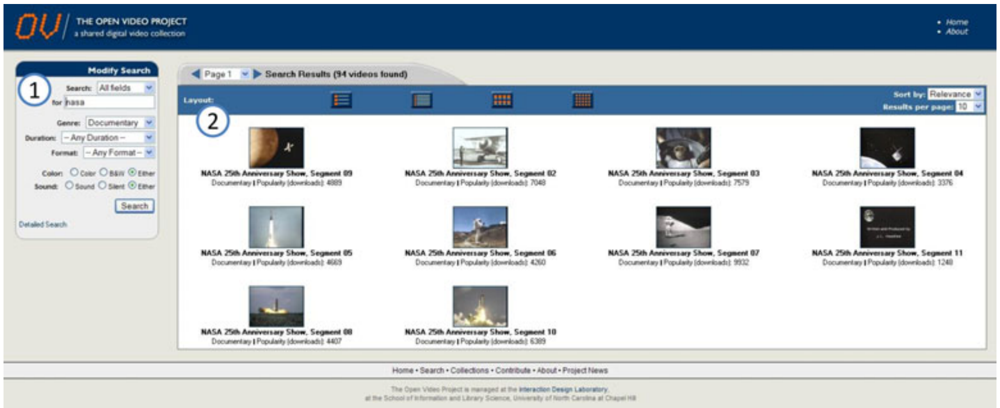
In one of the earlier efforts for supporting video retrieval, Arman et al. [57] proposed the use of the concept of key frames (denoted as Rframes in their paper), which are representative frames of shots, for chronological browsing of the content of a video sequence. Their approach uses simple motion analysis to find shot boundaries in a video sequence. For every shot a key frame is selected by using shape and color analysis. In addition to chronological browsing of key frames, their approach already allows selecting a key frame and searching for other similar key frames in the video sequence. For visualization of the results, they proposed that good results be displayed in original size (e.g., 100%), somewhat similar results in a smaller size (e.g., 33%), and bad results in an even smaller size (e.g., 5%). Several other papers have been published that use key-frame-based browsing of shots in a video sequence, usually by showing a page-based grid-like visualization of key frames (this is also called Storyboard) [58–67]. Some of them propose clustering of key frames into a hierarchical structure [58,60,63,65]. Considering the large amount of systems that visualize search results in a storyboard view, this approach can be seen as the standard visualization method. In the remainder of this section, we survey a few representative interfaces that rely on this visualization paradigm. An introduction on different paradigms is given by Christel [68].

Table 1 Overview of video-player-like video browsing applications.

	Browsing/ Querying	Input Files	Unit Structure	Video Domain	Content Analysis	Visualization/Interaction
Li et al. [28]	yes/no	1	frame	all	audio/speech analysis (pause removal), shot boundary detection, text recognition	similar to a video player with speeded-up playback and navigation indices
Barbierei et al. [29] (ColorBrowser)	yes/no	1	frame	all	dominant color, audio-track volume	colored seeker bar
Tang et al. [30] (NewsEye)	yes/yes	1	story	news	unsupervised clustering techniques for shot boundary detection and story segmentation, OCR	similar to a video player with advanced navigation helps and caption text display/search
Divakaran [31]	yes/no	1	frame	all	MPEG-7 motion activity	adaptive fast playback
Peker et al. [32]	yes/no	1	frame	all	temporal frequency and spatiotemporal complexity based on DCT block histograms	adaptive fast playback
Liu et al. [33] (NewsBR)	yes/yes	1	story	news	shot boundary detection (χ^2), silence detection, sobel filtering, OCR	similar to a video player with advanced navigation helps and caption text search
Moraveji et al. [34] (Color Bars)	yes/no	1	frame	all	text-based annotation	2D visualization through color bars as a seeker bar
Hürst et al. [35,36] (Zoom Slider)	yes/no	1	frame (time)	all	not required	common video player
Divakaran [40] (PVR)	yes/no	1	frame	sports	audio volume analysis	2D audio volume plot as a seeker bar

Table 2 Overview of video-player-like video browsing applications (cont'd).

	Browsing/ Querying	Input Files	Unit Structure	Video Domain	Content Analysis	Visualization/Interaction
Dragicevic et al. [41] (dimP)	yes/no	1	frame	all	optical motion flow estimation with SIFT	flow dragging based on optical flow estimation
Kimber et al. [43]	yes/no	1	frame	surveillance	foreground/background segmentation and object tracking	flow dragging based on optical flow estimation
Chen et al. [44] (EmoPlayer)	yes/no	1	scene	videos containing emotions	manual annotations	colored seeker bar
Chang et al. [45] (Smart Video Player)	yes/no	1	shot	all	shot boundary detection, annotation-based similarity analysis of shots	filmstrip view of key frames based on a user-selected level of detail, recommendation function
Drucker et al. [46] (SmartSkip)	yes/no	1	frame	all	not required	filmstrip view of key frames based on a user-selected level of detail
Liu and Zhang [47] (SportBR)	yes/yes	1	frame	soccer	color layout, speech and text recognition	video player with additional features for navigation and text-based search
Vakkalanka et al. [48] (NVIBRS)	yes/yes	1	story	news	shot boundary detection, motion analysis, face detection	news browsing by a tree of news story units
Rehatschek et al. [49] and Bailer et al. [50] (SVAT)	yes/yes	1	frame	all	shot boundary detection	temporal view of stripe images, key frames and motion events; content-based similarity search
Schoeffmann et al. [51–53] (Video Explorer)	yes/no	1	frame	all	dominant color extraction, motion analysis (motion vector classification)	interactive navigation summaries visualizing the temporal flow of dominant colors and motion characteristics
Cheng et al. [54] (SmartPlayer)	yes/no	1	shot	all	shot boundary detection based on color histograms, optical flow analysis	"scenic car driving" representation and automatic playback speed adaptation


OV
THE OPEN VIDEO PROJECT
a shared digital video collection
• Home
• About
Modify Search
1
2
Layout:
Detailed Search
Home • Search • Collections • Contribute • About • Project News

Search: All fields
for nasa
Genre: Documentary
Duration: -- Any Duration --
Format: -- Any Format --
Color: ○ Color ○ B&W ● Either
Sound: ○ Sound ○ Silent ● Either
Search

Page 1 Search Results (94 videos found)

Sort by: Relevance
Results per page: 10

NASA 25th Anniversary Show, Segment 09
Documentary | Popularity (downloads): 4889

NASA 25th Anniversary Show, Segment 02
Documentary | Popularity (downloads): 7048

NASA 25th Anniversary Show, Segment 03
Documentary | Popularity (downloads): 7579

NASA 25th Anniversary Show, Segment 04
Documentary | Popularity (downloads): 3376

NASA 25th Anniversary Show, Segment 05
Documentary | Popularity (downloads): 4669

NASA 25th Anniversary Show, Segment 06
Documentary | Popularity (downloads): 4260

NASA 25th Anniversary Show, Segment 07
Documentary | Popularity (downloads): 9932

NASA 25th Anniversary Show, Segment 11
Documentary | Popularity (downloads): 1248

NASA 25th Anniversary Show, Segment 08
Documentary | Popularity (downloads): 4407

NASA 25th Anniversary Show, Segment 10
Documentary | Popularity (downloads): 6389

The Open Video Project is managed at the Interaction Design Laboratory,
at the School of Information and Library Science, University of North Carolina at Chapel Hill

Fig. 8 Open video graphical user interface (screenshot taken from online system).

The first efforts to provide a digital library started in 1996. The researchers from the University of North Carolina at Chapel Hill indexed short video segments of videos and joined them with images, text, and hyperlinks in a dynamic query user interface. Their project has evolved since then so that now, digitalized video clips from multiple sources are combined into the Open Video Project [69]. Figure 8 shows a screenshot of the actual interface. It allows a textual search to be triggered by entering a query, denoted as (1) in the screenshot, and the possibility of browsing through the collections. Results are listed based on their importance to the given search query, denoted as (2) in the screenshot.

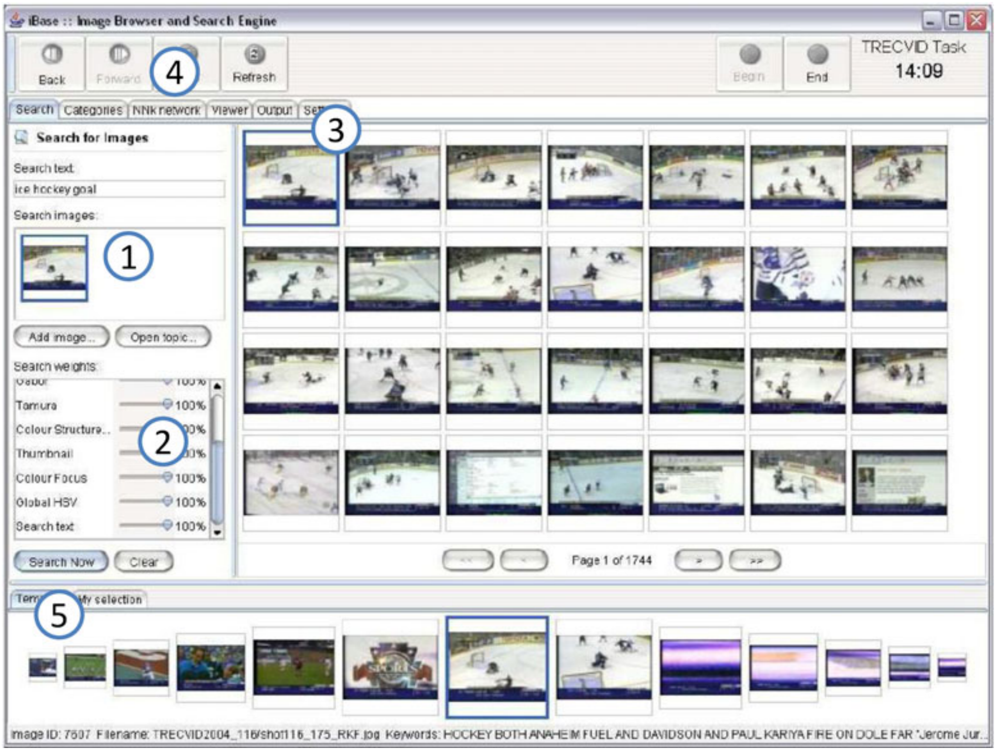
Deng and Manjunath [70] introduce a system using low-level visual features for content-based search and retrieval. Their system is based on shots and uses automatic shot partitioning and low-level feature extraction from compressed and decompressed domains. More specifically, videos are indexed using 256-bin RGB color histograms, motion histograms computed from MPEG motion vectors, and Gabor texture information. By giving an example shot, their system is able to retrieve similar shots of a video according to the three mentioned low-level features. The user may change the weights of the similarity matching for each of the three features.

Komlodi et al. [61,71] revealed in their user study that key-frame-based approaches such as the storyboards are still the preferred methods for seeking, even if additional time is required to interact with the user interface (scroll bars) and for eye movements. Dynamic approaches such as slideshows often display the content with a fixed frame rate and don't allow the user to adjust it.

An alternative approach to the linear storyboard navigation is to present key frames in a layered/hierarchical manner [65]. At the top level, a single key frame represents the entire video, whereas the number of key frames is increased at each level. If additional semantic information was extracted (e.g., an importance score), key frames may be displayed in different sizes, drawing the user's attention to important key frames in the first place [59,64]. These scores can also be applied to dynamic approaches to adjust the playback speed and skip unimportant scenes.

From 1998 to 2001, INRIA and Alcatel Alstom Research (AAR) developed the VideoPrep system, which allows automatic shot, key-frame, object, and scene segmentation. The corresponding viewer VideoClic is able to provide direct linking between, e.g., the same objects found on different temporal places. Some details about that work can be found in Ref. 72, pp. 20–23.

With the CueVideo project, Srinivasan et al. [73] have presented a browsing interface that allows several visualizations of the video content. Their system is based on shots and consists of visual content presentation, aural content presentation, and technical statistics. Visual content presentation comprises (1) a storyboard where for each shot a key frame is presented, and (2) a motion storyboard where for each shot an animated image is presented. The audio view shows a



Search text: ice hockey goal

Search images:

Search weights:

Gabor 100%
Tamura 100%
Colour Structure... 100%
Thumbnail 100%
Colour Focus 100%
Global HSV 100%
Search text 100%

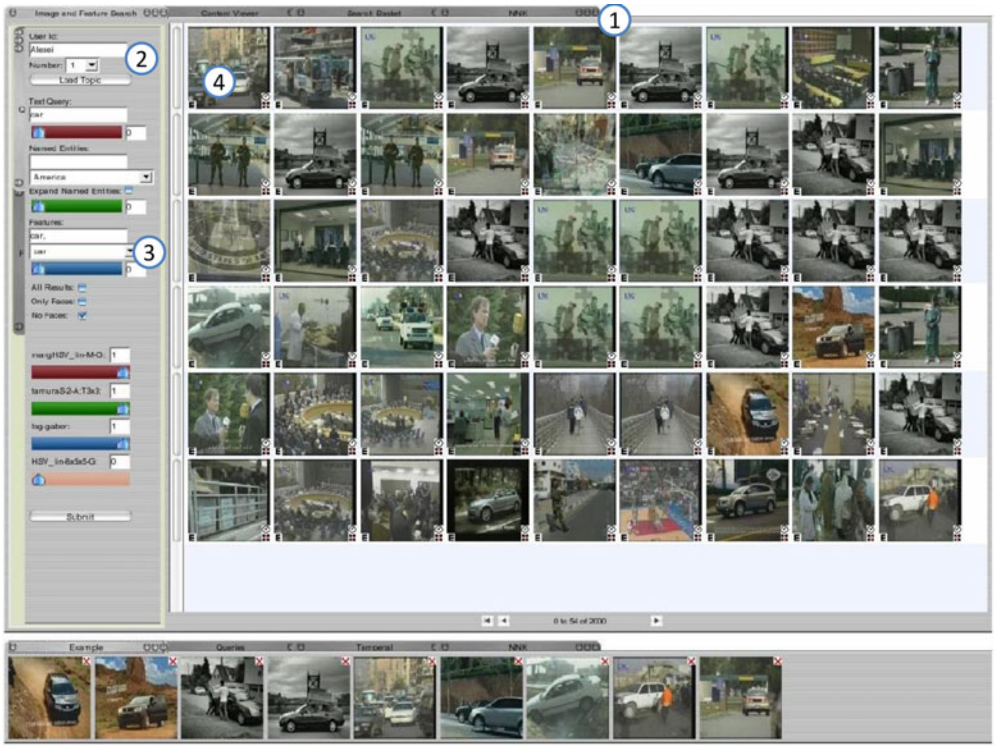
Image ID: 7507 Filename: TRECVID2004_116/shot116_175_RKF.jpg Keywords: HOCKEY BOTH ANAHEIM FUEL AND DAVIDSON AND PAUL KARIYA FIRE ON DOLE FAR 'Jerome Jur...

Fig. 9 Video browsing/retrieval as proposed by Heesch et al. [74].

classification of the audio tracks into the categories music, speech, and interesting audio events. In a user study they found out that the most popular view was the storyboard view, which is a similar result as already found by Komlodi et al. [61,71]. Users criticized the miss of the "top 10 key frames" and the bad scaling of the storyboard for long videos, but found it helpful (for content comprehension) to have different views.

Heesch et al. [74] presented a tool for video retrieval and video browsing (Fig. 9), which they have used for TRECVID. [75]. The tool allows a video to be searched and browsed in different dimensions in a storyboard manner. A user can (1) select an image (or key frame of a shot) as input. This image is further used by a feature-based search (2) that uses a feature vector consisting of nine different features for comparison (in general, color, texture, and transcript text). A user can manually tune the weighting of the different features. In the right part of the window, the results of the search are presented in a line-by-line and page-by-page manner (3). The best result is presented at the left-top position of the first page and the worst result is presented at the right-bottom position of the last page. Furthermore, they use a relevance feedback technique in order to improve repeated search. On another tab [called NN^k network, (4)], the nearest neighbors of a selected image can be shown in a graph-like visualization. To provide temporal browsing they also use a fish-eye visualization at the bottom of the window (5) in which the image of interest (selected on any view) is always shown in the center.

An extension of this approach is introduced by Ghoshal et al. [76]. Their interface, shown in Fig. 10, is split into two main panels with the browsing panel taking up to 80% of the screen. The browsing tab (1) is divided into four tabs that provide different categories: Image feature search, content viewer, search basket, and NN^k key-frame browsing. In the image & feature search tab (2), users can enter free text, named entities, and visual concepts. Besides, they can specify the weighting of each textual and visual feature using a sliding bar (3). The content viewer tab is divided into two tabs. On the left-hand side (4), textual metadata of the last clicked key frame is presented, while on the right-hand side, the full key frame is shown. In the search basket tab,



User Id: Alesei
Number: 1
Text Query: car
Named Entities:
America
Expand Named Entities: □
Features: car,
car
All Results: □
Only Faces: □
No Faces: ☑
margHSV_lin-M-G: 1
tamuraS2-A:T3x3: 1
log gabor: 1
HSV_lin-8x5x5-G: 0

Fig. 10 Video browsing/retrieval as proposed by Ghoshal et al. [76].

key frames that are currently selected are displayed. The NN^k browsing tab shows these 30 key frames that are nearest to the last clicked key frame in the visual feature space.

Rautiainen et al. [77] studied content-based querying enriched with relevance feedback by introducing a content-based query tool. Their retrieval system supports three different querying facilities: query by textual keyword, query by example, and query by concept. The interface, shown in Fig. 11, provides a list of semantic concepts a user can choose from. Textual-based queries can be added in a text field on the top left-hand side of the interface. Retrieved shots are represented as thumbnails of key frames, together with the spoken text in the most dominant part of the interface. By selecting key frames, users can browse the data collection using a cluster-based browsing interface [78]. Figure 12 shows a screenshot of this interface. It is divided into two basic parts. On top is a panel displaying the selected thumbnail and other frames of the video in chronological order (1). The second part displays similar key frames that have been retrieved by multiple content-based queries based on user-selected features (2). The key frames are organized in parallel order as a similarity matrix, showing the most similar matches in the first column. This enables the user to browse through a timeline and see similar shots at the same time. Each transition in the timeline will automatically update the key frames in the similarity matrix.

Campbell et al. [79] introduced a web-based retrieval interface. Using this interface, users can start a retrieval based on visual features, textual queries, or concepts. Figure 13(a) shows an example retrieval result. The interface provides functionalities to improve the visualization of retrieved key frames by grouping them into clusters according to their metadata, such as video name or channel. Figure 13(b) shows an example grouping.

A similar approach is studied by Bailer et al. [80]. In their interface, shown in Fig. 14, retrieval results are categorized into clusters. Single key frames represent each cluster in the result list (1). Controls around the panel (2) depicting the search results allow the users to resize the presentation of these key frames and to scroll through the list.

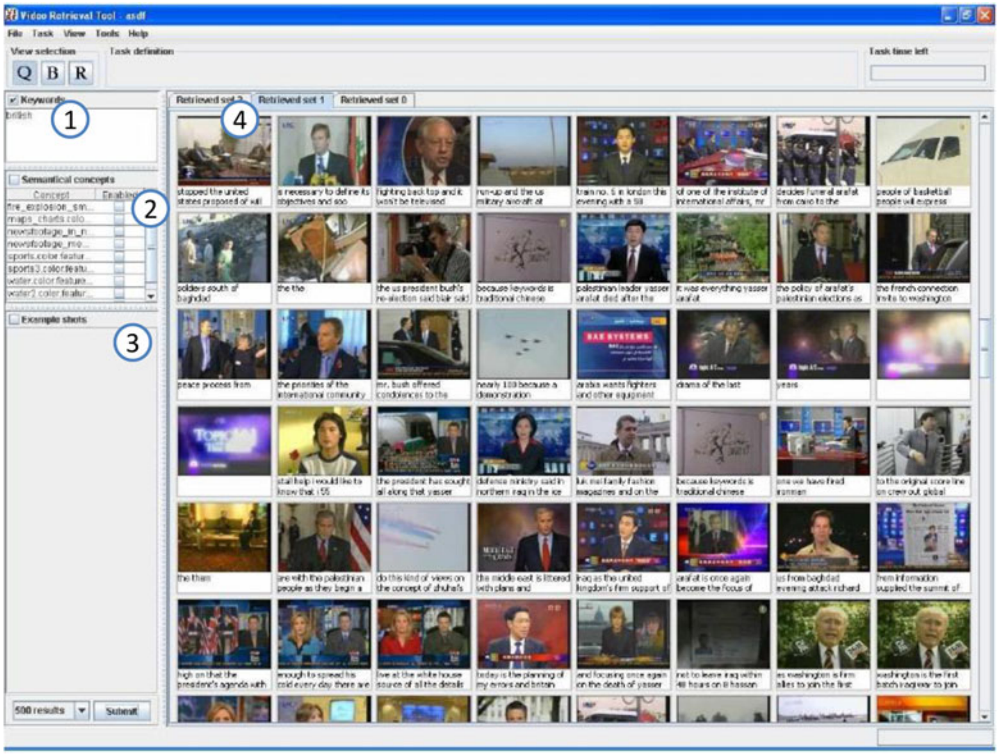


Fig. 11 The content-based query tool as proposed by Rautiainen et al. [77].

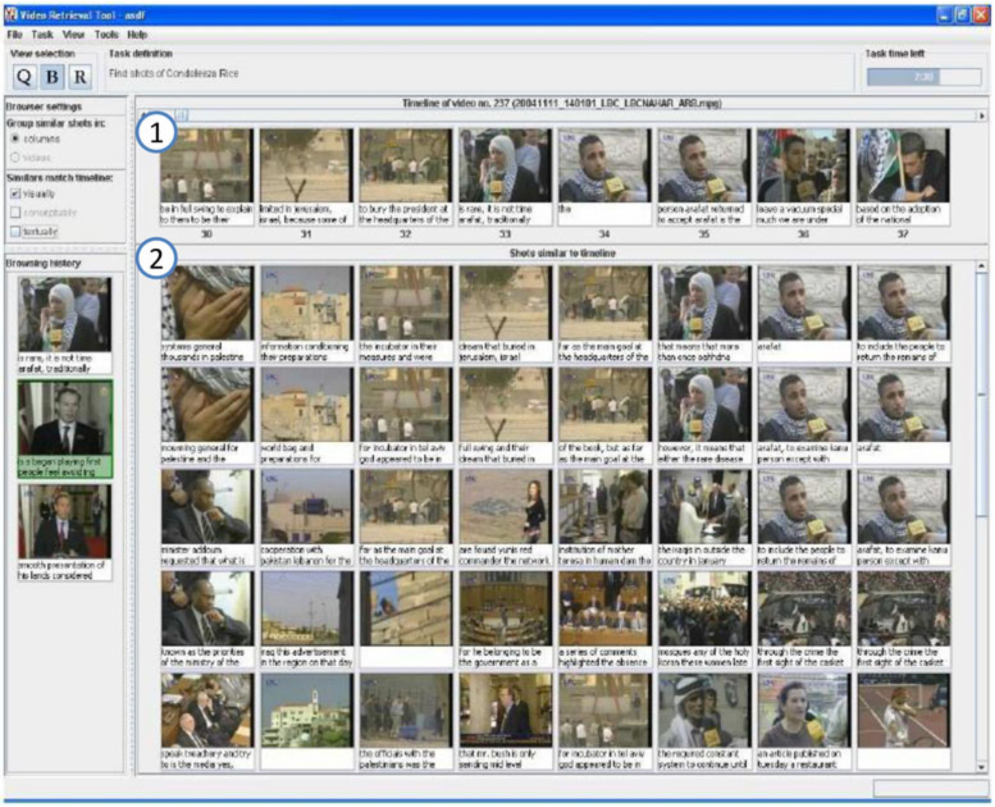
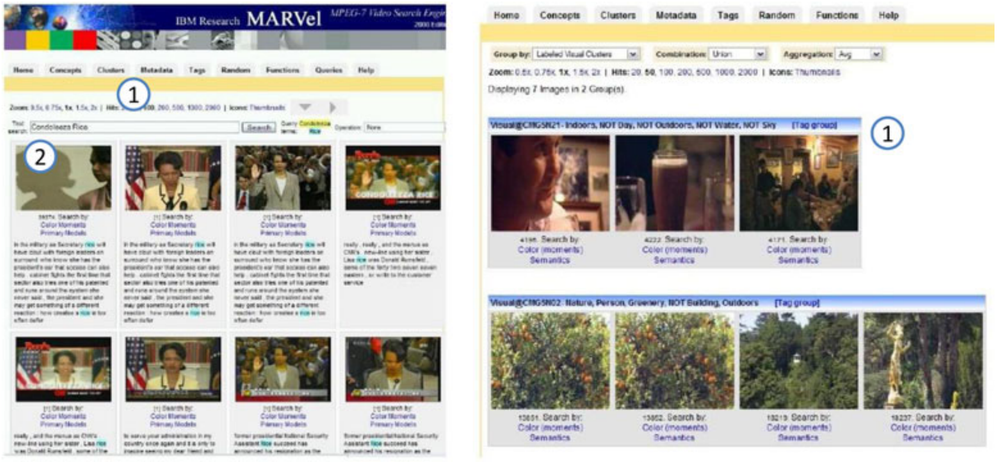


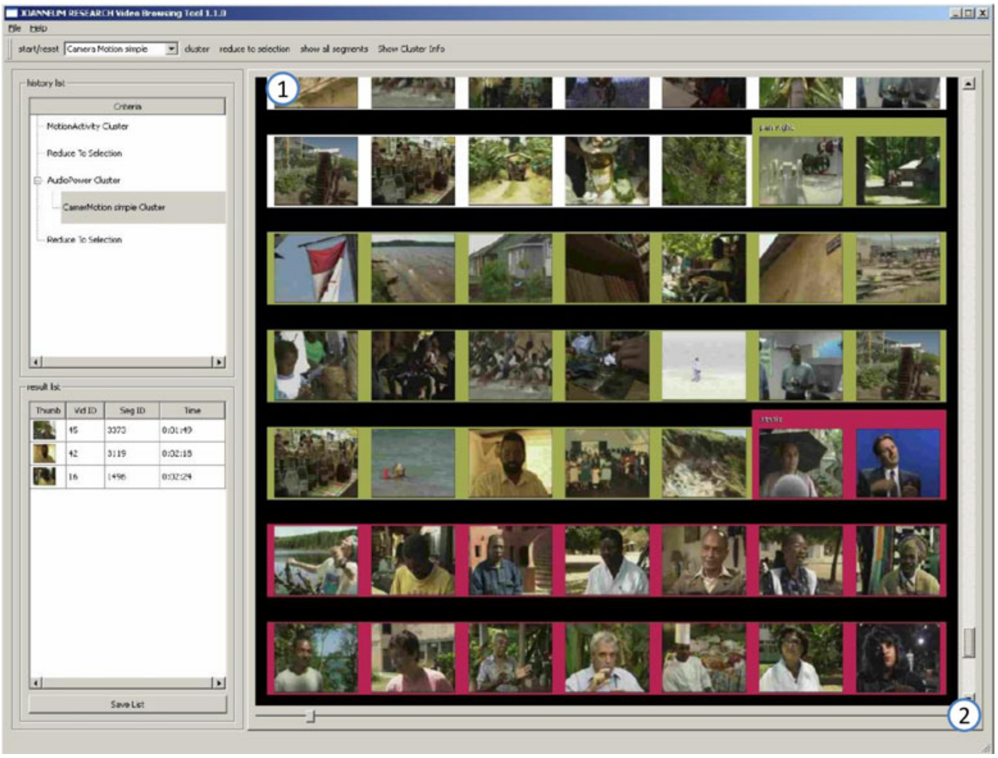
Fig. 12 The cluster-based query tool as proposed by Rautiainen et al. [77].


IBM Research MARVel
MPEG-7 Video Search Engine
Home
Concepts
Clusters
Metadata
Tags
Random
Functions
Queries
Help
Condoleezza Rice
Search
1
2
Group by: Labeled Visual Clusters
Combination: Union
Aggregation: Avg
Zoom: 0.5x, 0.75x, 1x, 1.5x, 2x | Hits: 20, 50, 100, 200, 500, 1000, 2000 | Icons: Thumbnails
Displaying 7 images in 2 Group(s)
Visual@CMGSN21: Indoors, NOT Day, NOT Outdoors, NOT Water, NOT Sky [Tag group]
Visual@CMGSN02: Nature, Person, Greenery, NOT Building, Outdoors [Tag group]
Search by: Color (moments) Semantics

(a) Interactive Search Example

(b) Grouping using Clusters

Fig. 13 (a) IBM MARVel used for interactive search, and (b) search results grouped by visual clusters. [79].



start/reset Camera Motion simple

Criteria
- MotionActivity Cluster
- Reduce To Selection
- AudioPower Cluster
  - CamerMotion simple Cluster
- Reduce To Selection

Thumb	Vid ID	Seg ID	Time
	45	3373	0:01:49
	42	3119	0:02:18
	16	1496	0:02:24

Fig. 14 Video browsing tool as proposed by Bailer et al. [80].

Foley et al. [81] experimented in collaborative retrieval by introducing a multiple-user system on a DiamondTouch [82] tabletop device. Using the interface, a user can add key frames as part of a search query and select which features of the key frame shall be a reference for similar results. In their experiment, they asked 16 novice users, divided into eight pairs, to perform various search tasks. Each pair was sitting around the tabletop, facing each other. An additional monitor was used for video playback. Figure 15 shows a screenshot of the interface. It provides facilities to enter a search query (1), browse through key frames (2), play a video shot (3),

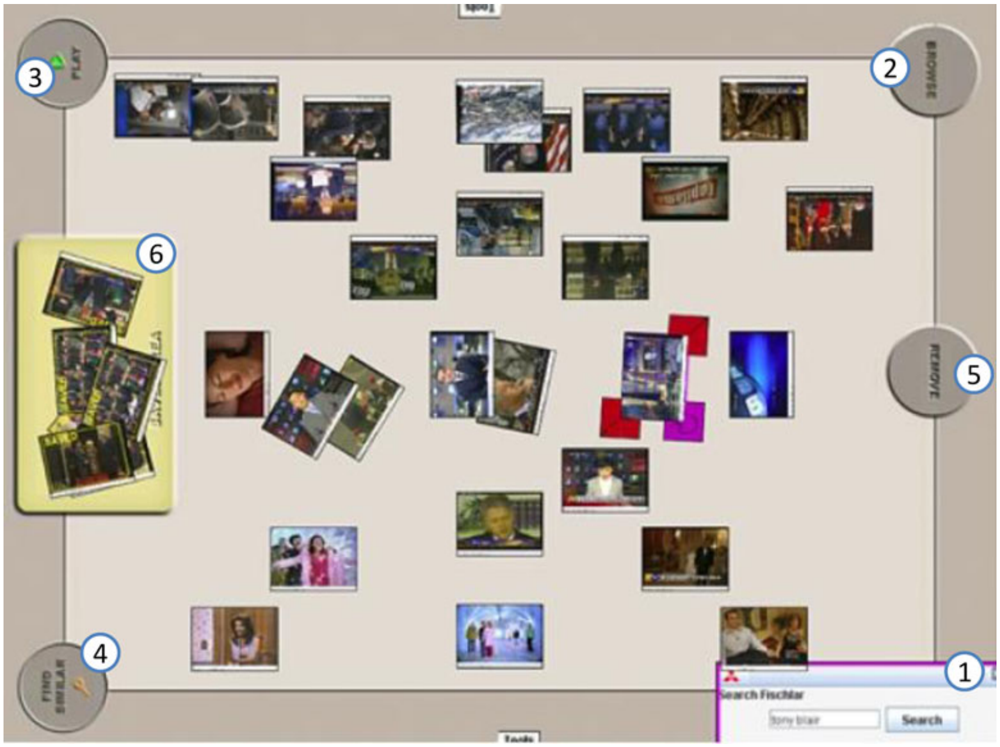


Fig. 15 Fischlar-DT system screenshot by Foley et al. [81].

find similar key frames (4), mark key frames as nonrelevant, (5) and save the key frames as a result (6).

Holthe and Ronningen [83] presented a video browsing plug-in for a Web browser, which can use hardware-accelerated graphics, if available, for improved browsing of video search results. Their browser allows compact views of preview images (in a 3D perspective) in order to increase the number of search results presentable on a single screen. When moving the mouse over a preview image the user can either zoom in or out on the image or start playback for the corresponding video segment, whereas the started video is presented in an overlay manner with the option of semitransparent display.

Villa et al. presented the FacetBrowser [84], a Web-based tool that allows the user to perform simultaneous search tasks within a video. A similar approach is introduced by Hopfgartner et al. [85]. The idea behind it is to enable a user to explore the content of a video by individual and parallel (sub)queries (and associated search results) in a way of exploratory search. A facet in that context is modeled as an individual search among others. The tool extracts speech transcripts from shots of the video for textual search. The results of a query are shown in a storyboard view where, in addition, a list of user-selected relevant shots for a particular query is shown as well. Moreover, the interface allows the user to add/remove search panels, to spatially move search panels, and to reuse search queries already performed in the history of a session.

Halvey et al. [86] introduced ViGOR, a grouping-oriented interface for search and retrieval in video libraries. The interface, shown in Fig. 16, allows users to create semantic groups to help conceptualize and organize their results for complex video search tasks. The interface is split into two main panels. On the left-hand side, users can enter a textual search query (1) and browse through the retrieval results (2). These results, represented by key frames, can be dragged and dropped to the example shots area (3) and will then be used as a visual query. The right-hand side of the interface consists of a workspace.

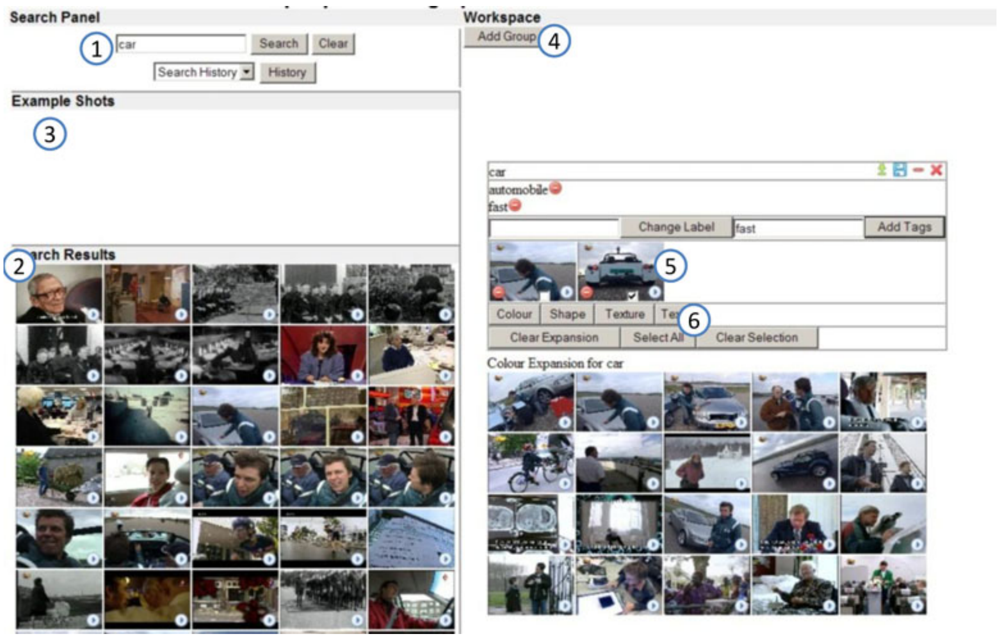


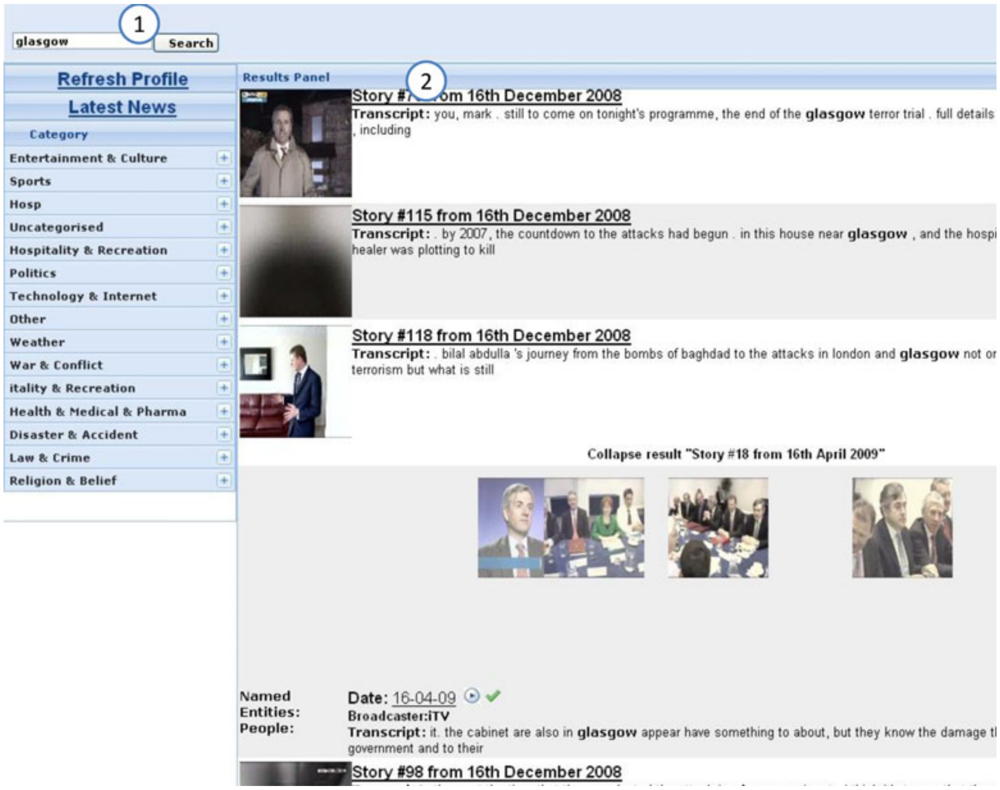
Fig. 16 ViGOR interface screenshot (taken from online demo).

In this workspace, users can create semantic groups (4), drag and drop key frames into these groups (5), and retrieve visually similar shots by exploiting various low-level visual features (6).

Adcock et al. [87] presented an interactive video search system called MediaMagic, which has been used for TRECVID [75] over several years. The shot-based system allows the user to search at textual, visual, and semantic levels. They use shot detection, color correlograms, and a support vector maching (SVM) to analyze the content. A rich search interface is provided, which enables text queries, image queries, and concept queries to be searched. In their interface they use visual clues to indicate which content item has been previously visited or explicitly excluded from search. Moreover, their system allows a multiple-user collaborative search to be performed.

Neo et al. [88] introduced an intuitive retrieval system called VisionGO that is optimized for a very fast browsing of the video corpus. The retrieval can be triggered by entering a textual search query. Furthermore, they can use keyboard shortcuts to quickly scroll through the retrieval results and/or to provide relevance feedback. The search query of later iterations is then further refined based on this feedback.

Most systems that have been introduced in this section support users in retrieving shots of a video. While this approach is useful in some cases, shots are not the ideal choice in other cases. Boreczky et al. [89] argue, for instance, that television news consists of a collection of story units that represent the different events that are relevant for the day of the broadcast. An example story unit from the broadcasting news domain is a report on yesterday's football match, followed by another story unit about the weather forecast. Various systems have been introduced to provide users access to news stories (e.g., Lee et al. [90], Pickering et al. [91], and Hopfgartner et al. [92]). In all cases, stories are treated as a series of shots and the corresponding key frames are visualized to represent a story. Figure 17 illustrates a representative interface as introduced by Hopfgartner and Jose [93]. Users can type in a search query and search results are ranked in either chronological order or based on their relevance to the search query.



Story #7 om 16th December 2008
Transcript: you, mark . still to come on tonight's programme, the end of the **glasgow** terror trial . full details , including

Story #115 from 16th December 2008
Transcript: . by 2007, the countdown to the attacks had begun . in this house near **glasgow** , and the hospi healer was plotting to kill

Story #118 from 16th December 2008
Transcript: . bilal abdulla 's journey from the bombs of baghdad to the attacks in london and **glasgow** not or terrorism but what is still

Collapse result "Story #18 from 16th April 2009"

Named Entities: People:

Date: 16-04-09 ✓
Broadcaster:iTV
Transcript: it. the cabinet are also in **glasgow** appear have something to about, but they know the damage tl government and to their

Story #98 from 16th December 2008

Fig. 17 Representative news video search interface (screenshot taken from live demo).

A summary of all introduced video retrieval interfaces is given in Table 3.

4 Video Browsing Applications Based on Video Surrogates and Unconventional Visualization

Many papers can be found in the literature [94–114] that describe video surrogates, which are alternative representations of the video content. The main purpose of video surrogates is to more quickly communicate the content of a video to the human observer. It is often used as a preview version for a video and should help the observer to decide whether the content is interesting or not. While such alternative representations are obviously important for video summarization, many proposals have been made to use video surrogates also for video browsing and navigation [94]. In this section we review applications using video surrogates for improved browsing or navigation.

The Mitsubishi Electric Research Laboratories (MERL) proposed several techniques for improved navigation within a video by novel content presentation. For example, the squeeze layout and the fish-eye layout [95] have been presented for improved fast-forward and rewind with personal digital video recorders (see Fig. 18). Both layouts extract future and past DC images of the MPEG stream. In addition to the current frame, the squeeze layout shows two DC images at normal size, one taken 30 sec in the future and another one taken 5 sec in the past, and squeezes together the other frames in between. The fish-eye layout shows gradually scaled DC images (in the future and in the past) next to the current frame, which is shown at normal size. Their evaluation has shown that subjects were significantly more accurate at fast-forward and rewind tasks with this display technique in comparison to a common VCR-like control set. In another paper of Wittenburg et al. [96] the visualization technique has been generalized to rapid serial visual presentation (RSVP). Their model defines spatial layouts of key frames in a

Table 3 Overview of video retrieval applications.

	Browsing/ Querying	Input Files	Unit Structure	Video Domain	Content Analysis	Visualization/Interaction
Open Video Project Geisler [69]	yes/yes	no limit	shot	all (c2)	shot boundary detection, text recognition	storyboard
Deng and Manjunath [70]	yes/yes	no limit	shot	all (c2)	shot boundary detection, text recognition	storyboard
Komlodi et al. [61]	yes/yes	no limit	shot	all (c2)	shot boundary detection, text recognition	storyboard
CueVideo [73]	yes/yes	no limit	shot	news (c2)	shot boundary detection, text recognition	motion storyboard
Heesch et al. [74]	yes/yes	no limit	shot	news	shot boundary detection, text recognition, visual retrieval	fish-eye visualization, storyboard
Rautiainen et al. [77]	yes/yes	no limit	shot	news	shot boundary detection, text recognition, visual retrieval	storyboard
Campbell et al. [79]	yes/yes	no limit	shot	news	shot boundary detection, text recognition, visual retrieval, concept filtering	storyboard, automatic grouping in clusters
Bailer et al. [80]	yes/yes	no limit	shot	news	shot boundary detection, text recognition, visual retrieval, concept clustering	storyboard, automatic grouping in clusters
Foley et al. [81]	yes/yes	no limit	shot	news	shot boundary detection, text recognition, visual retrieval	DiamondTouch
Villa et al. [84] (*FacetBrowser*)	yes/yes	no limited	shot	news	shot boundary detection, text recognition, visual retrieval	Facetted browsing
Halvey et al. [86] (*ViGOR*)	yes/yes	no limit	shot	news	shot boundary detection, text recognition, visual retrieval	storyboard, manual grouping
Adcock et al. [87] (*Media Magic*)	yes/yes	no limit	shot	news	shot boundary detection, color correlograms (with SVM)	storyboard, video player component, visual cues
VisionGo [88]	yes/yes	no limit	shot	news	shot boundary detection, text recognition, visual retrieval	designed for quick access
Hopfgartner and Jose [93]	yes/yes	no limit	story	news	story boundary detection, text recognition	storyboard, fish-eye visualization of story shots

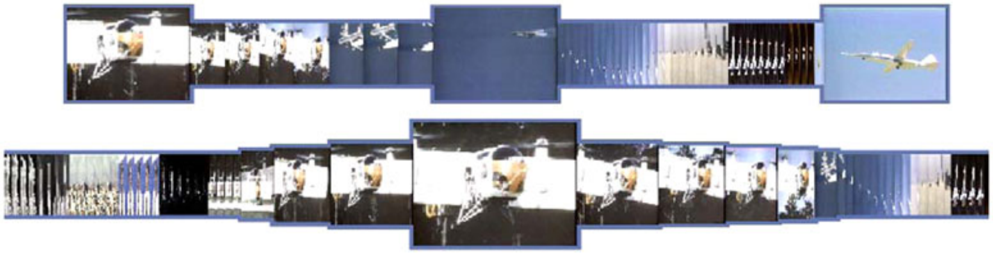


Fig. 18 The squeeze and fish-eye layouts for improved fast-forward and rewind. [95] © 2007 IEEE.

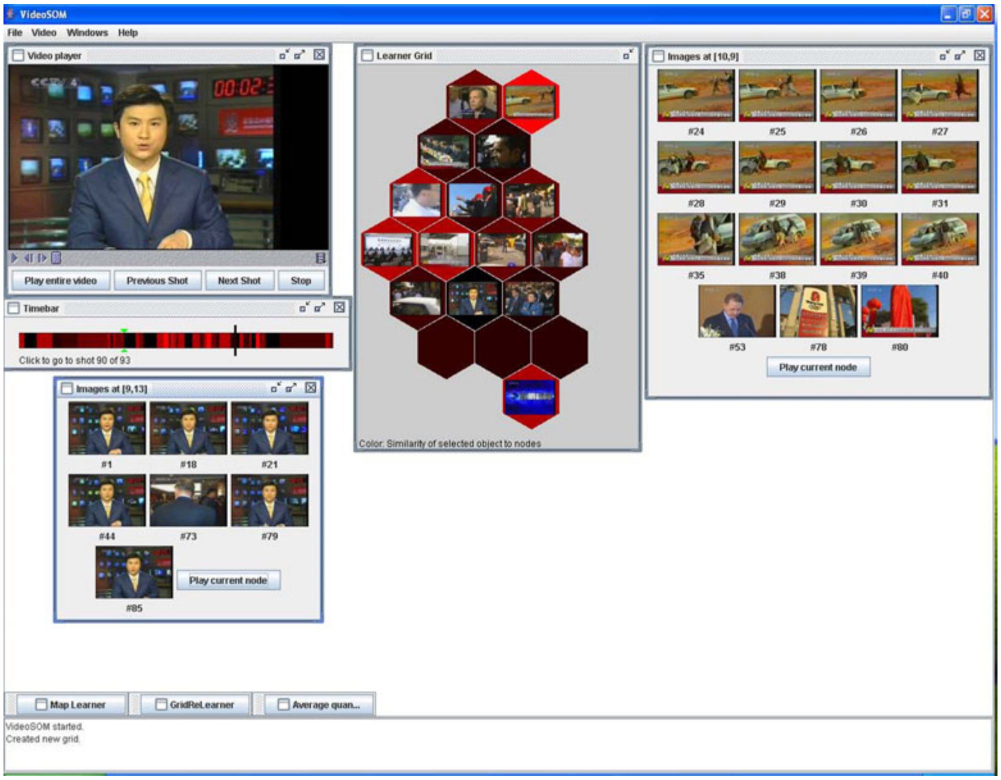
3D-like manner in different variations of trajectories. They evaluated the proposed RSVP technique for the purpose of video browsing by a user experiment with 15 subjects to compare it to the traditional VCR-like navigation set. The subjects were asked to answer questions such as "Find the next commercials block." They showed that their approach can significantly outperform the VCR-like navigation set in accuracy. However, no significant difference was found in the task completion time. Shipman et al. [97] described how the techniques of Wittenburg et al. [96] have been adapted to a consumer product.

Campanella et al. [98,99] proposed a visualization of MPEG-7 low-level features (such as dominant color, motion intensity, edge histogram, etc.) consisting of three parts. The main part of the visualization consists of a Cartesian plane showing small squares representing shots, where each square is painted in the dominant color of the corresponding shot. The user can select a specific feature for both the x-axis and the y-axis, which immediately affects the positioning of those squares. For instance, motion intensity could be chosen for the y-axis whereas dominant color could be chosen for the x-axis (colors are ordered according to the hue value). That visualization scheme enables a user to detect clusters of shots and to determine the distances of such clusters, according to a particular feature. Below the main window the shots are visualized in a temporal manner by painting stripes in the dominant color of each shot. Additionally, the right side shows key frames of the currently selected shot. In a recent paper Campanella et al. [115] describe an extended version of their tool with more interaction possibilities and additional views.

Axelrod et al. [100] presented an interactive video browser that uses pose slices, which are instantaneous objects' appearances, to visualize the activities within shots. They perform a foreground/background segmentation in order to find the pose slices, which are finally rendered in a 3D perspective. Their video browser allows several positions of an object to be simultaneously shown in a single video playback. Furthermore, the application enables a user to interactively control the viewing angle of the visualization.

Hauptmann et al. [101] proposed the so-called extreme video retrieval (XVR) approach, which tries to exploit both maximal use of human perception skills and the systems' ability to learn from human interaction. The basic idea behind it is that a human can filter out the best results from a query and can tell the system which results were right and which ones were wrong (a kind of relevance feedback). Therefore, they developed a RSVP approach, where key frames of a query result are rapidly presented to the user who marks the correct results by pressing a key. By always presenting the key frame at the same spatial location, their system avoids eye movements and, thus, minimizes the time necessary for a user to perceive the content of an image. The frame rate, i.e., how fast the images are presented, is determined by the user. After the first run, a second correction phase with lower frequency is used to recheck marked key frames. From this basic principle, extended versions have been implemented, where up to 4×4 images can be presented at the same time (Stereo RSVP), based on the natural parallelism of human binocular vision. In that case the user has 16 keys to mark a correct image in a grid-like presentation.

Eidenberger [102] proposed a video browsing approach that uses similarity-based clustering. More precisely, a self-organizing map (SOM), which is a neural network that uses feed-forward



VideoSOM started.
Created new grid.

Fig. 19 Video browsing with VideoSOM. © 2006 Bärecke.

learning, is employed as a similarity-based clustering method. Visually similar segments of a video are grouped together and visualized in hierarchically organized index trees. He presented two types of index trees that can efficiently visualize the content of a video. While the time index tree shows the temporal structure of the video in a top-down manner, the content index tree shows the shot-based structure of the video in a bottom-up approach (i.e., the user starts browsing at a specific shot). The clusters are visualized as hexagonally shaped cells showing key frames of shots. The user can interactively select a certain cell and step one layer deeper in the hierarchical tree structure to see more details of the selected shot. The number of layers in the tree depends on the length of the video. The user is able to switch between both views at any time during the browsing process, which helps to preserve the browsing context. For the SOM-based clustering process several different types of MPEG-7 visual features, extracted for every frame, are used.

A similar idea has been presented by Bärecke et al. who also used a growing SOM (Fig. 19) to build a video browsing application called VideoSOM [103]. Shots are nontemporally clustered according to a (probably color) histogram. Their tool provides a video player and several additional views at a glance:

- a self-organizing map window, showing key frames of shot clusters gained through the learning phase,
- a list of shots (visualized by key frames) according to a selected cluster, and
- a timeline showing temporal positions of shots in the shot window.

Goeau et al. [104] proposed the so-called table of video contents (TOVC) for browsing story-based video content such as news, interviews, or sports summaries. Based on low-level features such as color histograms in different color spaces, gradient orientation histograms, and motion model estimation (based on corner detection), they compute a similarity matrix that is further used for visualization. The 2D visualization uses a video backbone, either identified in a supervised way by an expert or automatically by finding clusters that contain the most covering frames. Every story is painted as a loop of key frames originating from that backbone.

de Rooij et al. [105] and Snoek et al. [106] introduced the notion of video threads, where a thread is a sequence of feature-based similar shots from several videos in some specific order. They differentiate between

(1) visual threads having visual similarity,
(2) textual threads having similar textual annotations,
(3) semantic threads having semantically equivalent shots,
(4) time threads having temporal similarity,
(5) query result threads having similarity according to a query, and
(6) history threads consisting of shots the user has already visited.

Based on the above-mentioned video threads they have implemented several different visualization schemes. The RotorBrowser starts from an initial query result. The user can select a focal shot S that is displayed (at a bigger size) in the center of the screen. According to that focal shot S the RotorBrowser provides several navigation paths by showing (parts of) all the video threads that contain S in a star formation. As the RotorBrowser has been proven to be too overwhelming for nonexpert users, the CrossBrowser has been developed. The CrossBrowser only provides horizontal and vertical navigation. For instance, the time thread is visualized in the horizontal line while the visually similar shots of S are visualized in a vertical line. In the TRECVID 2006 evaluation [116] of mean average precision, the CrossBrowser placed second and the RotorBrowser placed sixth. The tool has been further extended in the ForkBrowser, which achieved even better results in the TRECVID 2008 evaluation. [117].

Adams et al. [107] published another interesting work called temporal semantic compression for video browsing. Their video browsing prototype (Fig. 20) allows shot-based navigation (bottom left in the figure), whereas only a few shots are shown at a glance containing the selected shot in the center. They compute a tempo function for every frame and every shot, based on camera motion (e.g., pan and tilt), audio energy, and shot length. The resulting function is plotted at the top right side of the window (not shown in the figure). Their prototype enables a user to individually select a "compression rate" in order to shorten (i.e., summarize) the video. This function can be used by a simple slider or by directly clicking into the playback area, whereas the compression rate is derived from the vertical position and, in addition, the playback time position is selected by the horizontal position. Moreover, several different compression modes can be chosen. While the linear compression mode simply speeds up playback, the midshot constant mode takes a constant amount from the middle of a shot at a constant playback rate. The pace-proportional mode uses a variable playback rate based on the frame-level tempo and the interesting-shots mode discards shots with low tempo values according to the selected compression rate.

Jansen et al. [108] recently proposed the use of VideoTrees (Fig. 21) as alternatives to storyboards. A VideoTree is a hierarchical tree-like temporal presentation of a video through key frames. The key frames are placed adjacently to their parents and siblings such that no edge lines are required to show the affiliation of a node. With each depth level the level of detail increases as well (until shot granularity). For example, a user may navigate from a semantic root segment to one of the subjacent scenes, then to one of the subjacent shot groups, and finally to one of the subjacent shots. The current selected node in the tree is always centered, showing the context (i.e., a few of the adjacent nodes) in the surrounding area. In a user study with 15 participants they showed that the VideoTrees can outperform storyboards in terms of search time (1.14 times faster). However, the study also revealed that users found the classical storyboard much easier and clear.

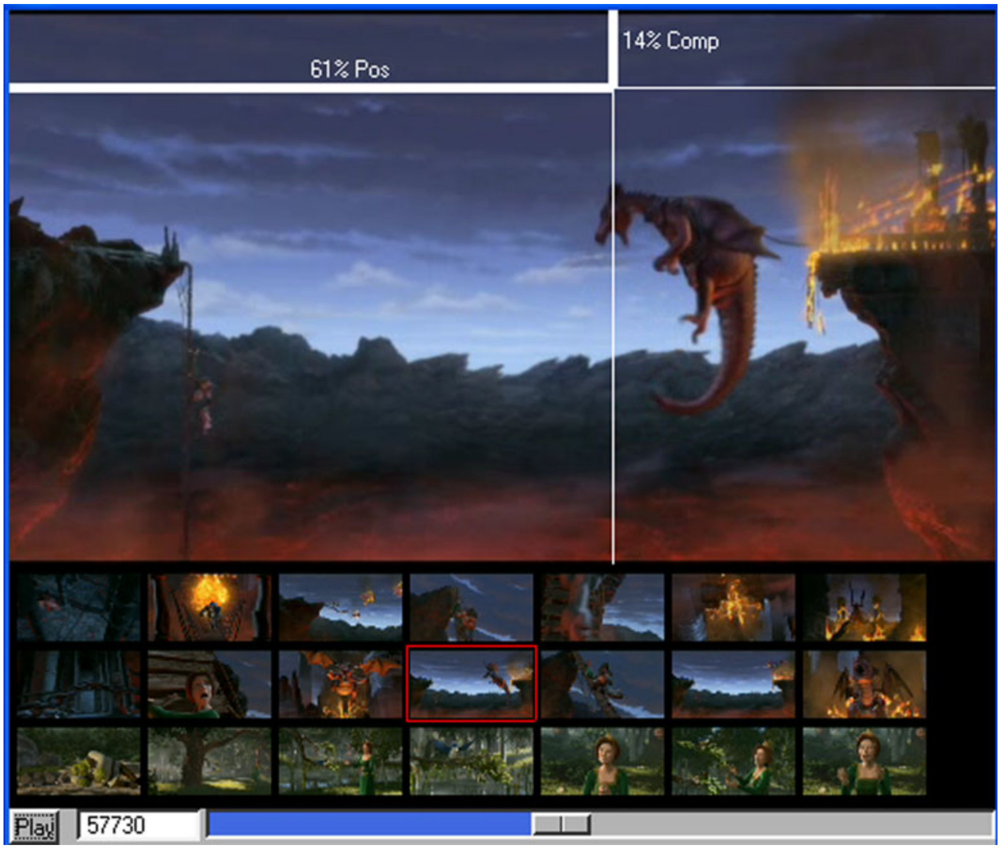


Fig. 20 Video browsing by temporal semantic compression. [107]

Table 4 gives an overview of the approaches reviewed in this section. All applications have been structured by the same criteria as used in Section 2.

5 Concluding Remarks

We have reviewed video browsing approaches that have been published in the literature within the last decade. We classified the existing approaches into three different types:

- interactively browsing and navigating through the content of a video in a video-player-like style,
- browsing the results of a video retrieval query (or a large video collection), and
- video browsing based on video surrogates.

Our review has shown that research in video browsing is very active and diverse. While a few approaches simply try to speed up the playback process for a video, several others try to improve the typical interaction model of a video player. In fact, the navigation features of a standard video player are still very similar to those of analog video cassette recorders invented in the 1960s (apart from faster random access). Even popular online video platforms use such primitive navigation functions. The main reason is surely that most users are familiar with the usage of simple video players. Section 2 has revealed that many other methods are available to improve common video players while keeping interaction simple. Many other approaches try to optimize the visual presentation of a large video collection or a number of search results. On one hand the storyboard has been established here as a standard means to display a large number of key frames and it is used in most video retrieval applications. On the other hand, Section 4 has shown that video surrogates can more effectively convey video content information.

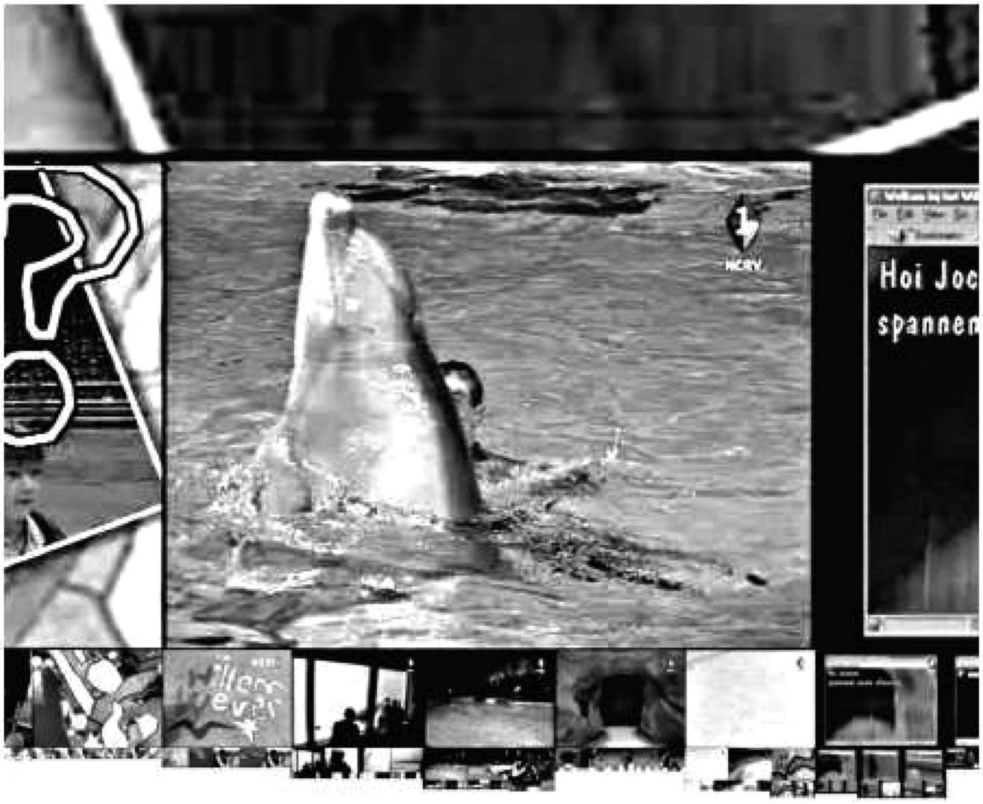


Fig. 21 Video browsing with the VideoTree. [108]

Appropriate video surrogates can significantly improve the performance of video browsing and video retrieval applications. The reason for this is that human users can easily and quickly identify content correlations from appropriate visualizations and use their personal knowledge and experience to improve the search process. Nevertheless, to design generally usable video surrogates might be difficult. It is obvious that video surrogates need to be specifically designed for several different types of video content and a great deal of research needs to be performed in that direction.

This review has not only shown that the user interfaces of video browsing applications are very diverse, but also that the methods used for content analysis are very different. While some methods use no content analysis at all, which has the non-negligible advantage of a short "start-up delay" for a new video from the user perspective, others perform intensive multimodel analysis. In general, we can conclude that the content analysis technique to be used is highly dependent on the video domain. For news videos, most approaches use text recognition and a few apply face detection. In contrast, motion and speech analysis is typically used for sports. If the application must be usable in several domains, color and motion features are often employed. The video domain also determines content segmentation. While shots are typically used for general-purpose applications, story units are the structuring element for news domain applications.

Future challenges are to further assist users in video browsing and exploration. Intelligent user interfaces are required that do not only visualize the video content but also adapt to the user. In a video retrieval scenario, this adaptation can be achieved by employing relevance feedback techniques. Moreover, considering the increasing amount of diverse user-generated video content, e.g., on social networking platforms, another challenge is how interfaces can deal with this low-quality material.

Table 4 Overview of video browsing applications using video surrogates.

	Browsing/ Querying	Input Files	Unit Structure	Video Domain	Content Analysis	Visualization/Interaction
MERL [95–97]	yes/no	1	frame	all	DC-image extraction from MPEG files	squeeze, fish-eye, RSVP
Campanella et al. [99,115]	yes/no	1	shot	all	dominant color, motion, temporal position of shots	interactive Cartesian plane
Axelrod et al. [100]	yes/no	1	shot	all	foreground/background segmentation	3D scene rendering of pose slices
Hauptmann et al. [101]	yes/no	1	shot	all	shot boundary detection	rapid serial visual presentation and manual browsing
Eidenberger [102]	yes/no	1	frame	all	shot boundary detection, MPEG-7 visual feature extraction (and similarity-based clustering with SOMs)	hierarchical browsing with self-organizing maps
Bärecke et al. [103] (VideoSOM)	yes/no	1	shot	all	shot boundary detection, clustering based on (color) histograms	self-organizing map browsing, storyboard, interactive shot-timeline, common playback
Goeau et al. [104] (TOVC)	yes/no	1	story	news	color histograms, gradient orientation histograms, motion model estimation	2D graph visualization (video backbone with loops)
de Rooij et al. [105] and Snoek et al. [106] (RotorBrowser, CrossBrowser)	yes/no	N	shot	all	shout boundary detection, visual cues, concepts	different visual browsing schemes for video threads
Adams et al. [107]	yes/no	1	shot	all	shout boundary detection, analysis of camera motion and audio energy	fast playback (time compression), interactive plots
Jansen et al. [108]	yes/no	1	shot	all	shout boundary detection	hierarchical navigation through shots and temporal shot groups

References

[1] G. Salton, *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*, Addison-Wesley Longman Publishing Co., Boston, MA, USA (1989).

[2] N. J. Belkin, P. G. Marchetti, and C. Cool, "Braque: design of an interface to support user interaction in information retrieval," *Inf. Process. Manage.* **29**(3), 325–344 (1993).

[3] A. Spink, H. Greisdorf, and J. Bateman, "From highly relevant to not relevant: examining different regions of relevance," *Inf. Process. Manage.* **34**(5), 599–621 (1998).

[4] M. Hearst, *Search User Interfaces*, Cambridge University Press, Cambridge, United Kingdom (2009).

[5] A. Jaimes, M. Christel, S. Gilles, S. Ramesh, and W.-Y. Ma, "Multimedia information retrieval: what is it, and why isn't anyone using it?," in *MIR '05: Proc. 7th ACM SIGMM Intl. Workshop on Multimedia Information Retrieval*, pp. 3–8, ACM Press, New York, NY, USA (2005).

[6] C. G. M. Snoek, M. Worring, D. C. Koelma, and A. W. M. Smeulders, "A learned lexicon-driven paradigm for interactive video retrieval," *IEEE Trans. Multimedia* **9**, 280–292 (Feb. 2007).

[7] M. Szummer and R. W. Picard, "Indoor-outdoor image classification," in *CAIVD '98: Proc. 1998 Intl. Workshop on Content-Based Access of Image and Video Databases (CAIVD '98)*, pp. 42–51, IEEE Computer Society, Washington, DC, USA (1998).

[8] A. Vailaya, M. A. T. Fiqueiredo, A. K. Jain, and H.-J. Zhang, "Image classification for content-based indexing," *IEEE Trans. Image Processing* **10**(1), 117–130 (2001).

[9] A. Tombros and M. Sanderson, "Advantages of query biased summaries in information retrieval," in *SIGIR '98: Proc. 21st Annual Intl. ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2–10, ACM, New York, NY, USA (1998).

[10] R. W. White, J. M. Jose, and I. Ruthven, "A task-oriented study on the influencing effects of query-biased summarisation in web searching," *Inf. Process. Manage.* **39**(5), 707–733 (2003).

[11] K. Otsuji, Y. Tonomura, and Y. Ohba, "Video browsing using brightness data," *Proc. SPIE*, **1606**, 980–989 (1991).

[12] Y. Nakajima, "A video browsing using fast scene cut detection for an efficient networked video database access," *IEICE Trans. Information Systems* **77**(12), 1355–1364 (1994).

[13] F. Arman, R. Depommier, A. Hsu, and M. Chiu, "Content-based browsing of video sequences," in *Proc. Second ACM Intl. Conference on Multimedia*, pp. 97–103, ACM New York, NY, USA (1994).

[14] H. Zhang, S. Smoliar, and J. Wu, "Content-based video browsing tools," *Proc. SPIE*, **2417**, 389–398 (1995).

[15] H. Zhang, C. Low, S. Smoliar, and J. Wu, "Video parsing, retrieval and browsing: an integrated and content-based solution," in *Proc. Third ACM Intl. Conf. Multimedia*, pp. 15–24, ACM (1995).

[16] M. Smith and T. Kanade, "Video skimming for quick browsing based on audio and image characterization," *Computer Science Technical Report*, Carnegie Mellon University (1995).

[17] M. Yeung, B. Yeo, W. Wolf, and B. Liu, "Video browsing using clustering and scene transitions on compressed sequences," in *Proc. SPIE*, **2417**, 399–414 (1995).

[18] D. Zhong, H. Zhang, and S. Chang, "Clustering methods for video browsing and annotation," in *Proc. SPIE*, **2670**, 239–246 (1996).

[19] M. Yeung, B. Yeo, and B. Liu, "Extracting story units from long programs for video browsing and navigation," in *Proc. Multimedia*, **1996**, 296–304 (1996).

[20] R. Zabih, J. Miller, and K. Mai, “Video browsing using edges and motion,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 439–446 (1996).

[21] R. Hjelsvold, R. Midtstraum, and O. Sandsta, “Searching and browsing a shared video database,” *Multimedia Database Systems*, 89–122 (1995).

[22] M. Yeung and B. Yeo, “Video visualization for compact presentation and fast browsing ofpictorial content,” *IEEE Trans. Circuits Systems Video Technol.* **7**(5), 771–785 (1997).

[23] B. Yeo and M. Yeung, “Classification, simplification, and dynamic visualization of scene transition graphs for video browsing,” *Proc. SPIE*, **3312**, 60–71 (1997).

[24] H. Zhang, J. Wu, D. Zhong, and S. Smoliar, “An integrated system for content-based video retrieval and browsing,” *Pattern Recognition* **30**(4), 643–658 (1997).

[25] I. Mani, D. House, and M. Maybury, “Towards content-based browsing of broadcast news video,” in *Intelligent Multimedia Information Retrieval*, M. T. Maybury, Ed. MIT Press, Cambridge, MA, pp. 241–258 (1997).

[26] D. Ponceleon, S. Srinivasan, A. Amir, D. Petkovic, and D. Diklic, “Key to effective video retrieval: effective cataloging and browsing,” in *Proc. Sixth ACM Intl. Conf. Multimedia*, pp. 99–107, ACM, New York, NY, USA (1998).

[27] A. F. Smeaton, P. Over, and W. Kraaij, “Evaluation campaigns and TRECVid,” in *MIR ‘06: Proc. 8th ACM Intl. Workshop Multimedia Information Retrieval*, pp. 321–330, ACM Press, New York, NY, USA (2006).

[28] F. Li, A. Gupta, E. Sanocki, L. He, and Y. Rui, “Browsing digital video,” in *Proc. SIGCHI Conf. Human Factors in Computing Systems*, pp. 169–176, ACM, New York, NY, USA (2000).

[29] M. Barbieri, G. Mekenkamp, M. Ceccarelli, and J. Nesvadba, “The color browser: a content driven linear video browsing tool,” *IEEE Intl. Conf. Multimedia and Expo, 2001*, pp. 627–630 (2001).

[30] X. Tang, X. Gao, and C. Wong, “NewsEye: a news video browsing and retrieval system,” in *Proc. 2001 Intl. Symp. Intelligent Multimedia, Video and Speech Processing, 2001*, pp. 150–153 (2001).

[31] A. Divakaran, K. Peker, R. Radhakrishnan, Z. Xiong, and R. Cabasson, “Video Summarization using MPEG-7 Motion Activity and Audio Descriptors,” Technical Report TR-2003-34, Mitsubishi Electric Research Laboratories (May 2003).

[32] K. Peker and A. Divakaran, “Adaptive fast playback-based video skimming using a compressed-domain visual complexity measure,” in *2004 IEEE Intl. Conf. Multimedia and Expo*, **3**, 2055–2058 (2004).

[33] J. Liu, Y. He, and M. Peng, “NewsBR: a content-based news video browsing and retrieval system,” in *Fourth Intl. Conf. Computer and Information Technology*, pp. 857–862 (2004).

[34] N. Moraveji, “Improving video browsing with an eye-tracking evaluation of feature-based color bars,” *Proc. 2004 Joint ACM/IEEE Conf. Digital Libraries*, pp. 49–50 (2004).

[35] W. Hürst, G. Gotz, and T. Lauer, “New methods for visual information seeking through video browsing,” in *Proc. Eighth Intl. Conf. Information Visualisation*, pp. 450–455 (2004).

[36] W. Hürst and P. Jarvers, “Interactive, Dynamic Video Browsing with the ZoomSlider Interface,” in *Proc. IEEE Intl. Conf. Multimedia and Expo*, pp. 558–561, IEEE, Amsterdam, The Netherlands (2005).

[37] W. Hürst, “Interactive audio-visual video browsing,” in *Proc. 14th Annual ACM Intl. Conf. Multimedia*, pp. 675–678, ACM, New York, NY, USA (2006).

[38] W. Hürst, G. Götz, and M. Welte, “A new interface for video browsing on PDAs,” in *Proc. 9th Intl. Conf. Human Computer Interaction with Mobile Devices and Services*, pp. 367–369, ACM, New York, NY, USA (2007).

[39] W. Hürst, G. Gotz, and M. Welte, "Interactive video browsing on mobile devices," in *Proc. 15th Intl. Conf. Multimedia*, **25**, 247–256 (2007).

[40] A. Divakaranand I. Otsuka, "A video-browsing-enhanced personal video recorder," in *14th Intl. Conf. Image Analysis and Processing Workshops*, pp. 137–142 (2007).

[41] P. Dragicevic, G. Ramos, J. Bibliowitcz, D. Nowrouzezahrai, R. Balakrishnan, and K. Singh, "Video browsing by direct manipulation," in *Proc. 26th Annual SIGCHI Conf. Human Factors in Computing Systems*, pp. 237–246, ACM, New York, NY, USA (2008).

[42] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Intl. J. Computer Vision* **60**(2), 91–110 (2004).

[43] D. Kimber, T. Dunnigan, A. Girgensohn, F. Shipman, T. Turner, and T. Yang, "Trailblazing: video playback control by direct object manipulation," in *IEEE Conf. Multimedia and Expo*, pp. 1015–1018 (2007).

[44] L. Chen, G. Chen, C. Xu, J. March, and S. Benford, "EmoPlayer: A media player for video clips with affective annotations," *Interacting with Computers* **20**(1), 17–28 (2008).

[45] L. Chang, Y. Yang, and X.-S. Hua, "Smart video player," in *IEEE Intl. Conf. Multimedia and Expo*, pp. 1605–1606 (2008).

[46] S. Drucker, A. Glatzer, S. De Mar, and C. Wong, "SmartSkip: consumer level browsing and skipping of digital video content," in *Proc. SIGCHI Conf. Human Factors in Computing Systems*, pp. 219–226, ACM, New York, NY, USA (2002).

[47] H. Liu and H. Zhang, "A content-based broadcasted sports video retrieval system using multiple modalities: SportBR," in *Fifth Intl. Conf. Computer and Information Technology*, pp. 652–656 (2005).

[48] S. Vakkalanka, S. Palanivel, and B. Yegnanarayana, "NVIBRS-news video indexing, browsing and retrieval system," in *Proc. 2005 Intl. Conf. Intelligent Sensing and Information Processing*, pp. 181–186 (2005).

[49] H. Rehatschek, W. Bailer, H. Neuschmied, S. Ober, and H. Bischof, "A tool supporting annotation and analysis of videos," S. Knauss and A.D. Ornella, Eds., *Reconfigurations: Interdisciplinary Perspectives on Religion in a Post-Secular Society*, LIT Verlag, Berlin, Münster, Wien, Zürich, London, ISBN 978-3-8258-0775-7, **3**, 253–268 (2007).

[50] W. Bailer, C. Schober, and G. Thallinger, "Video content browsing based on iterative feature clustering for rushes exploitation," in *Proc. TRECVid Workshop*, pp. 230–239 (2006).

[51] K. Schoeffmann and L. Boeszoermenyi, "Video browsing using interactive navigation summaries," in *Proc. 7th Intl. Workshop on Content-Based Multimedia Indexing*, IEEE, Chania, Crete (June 2009).

[52] K. Schoeffmann and L. Boeszoermenyi, "Enhancing seeker-bars of video players with dominant color rivers," in *Advances in Multimedia Modeling*, Y.-P. P. Chen, Z. Zhang, S. Boll, Q. Tian, and L. Zhang, Eds., Springer, Chongqing, China (January 2010).

[53] K. Schoeffmann, M. Taschwer, and L. Boeszoermenyi, "Video browsing using motion visualization," in *Proc. IEEE Intl. Conf. Multimedia and Expo*, IEEE, New York, USA (July 2009).

[54] K.-Y. Cheng, S.-J. Luo, B.-Y. Chen, and H.-H. Chu, "Smartplayer: user-centric video fast-forwarding," in *CHI '09: Proc. 27th Intl. Conf. Human Factors in Computing Systems*, pp. 789–798, ACM, New York, NY, USA (2009).

[55] A. Haubold and J. Kender, "VAST MM: multimedia browser for presentation video," in *Proc. 6th ACM Intl. Conf. Image and Video Retrieval*, pp. 41–48, ACM Press, New York, NY, USA (2007).

[56] R. C. Veltkamp, H. Burkhardt, and H.-P. Kriegel, Eds., *State-of-the-Art in Content-Based Image and Video Retrieval*, (Dagstuhl Seminar, 5-10 December 1999), Kluwer (2001).

[57] F. Arman, R. Depommier, A. Hsu, and M. Chiu, "Content-based browsing of video sequences," *Proc. Second ACM Intl. Conf. Multimedia*, pp. 97–103 (1994).

[58] D. Zhong, H. Zhang, and S. Chang, "Clustering methods for video browsing and annotation," *Proc. SPIE* **2670**, 239–246 (1996).

[59] M. Yeung and B.-L. Yeo, "Video visualization for compact representation and fast browsing of pictorial content," *IEEE Trans. Circ. Syst. Video Technol.* **7**(5), 771–785 (1997).

[60] H. J. Zhang, J. Wu, D. Zhong, and S. W. Smoliar, "An integrated system for content-based video retrieval and browsing," *Pattern Recognition* **30**(4), 643–658 (1997).

[61] A. Komlodi and G. Marchionini, "Key frame preview techniques for video browsing," *Proc. 3rd ACM Conf. Digital Libraries*, pp. 118–125 (1998).

[62] A. Komlodi and L. Slaughter, "Visual video browsing interfaces using key frames," in *CHI '98 Conference Summary on Human Factors in Computing Systems*, pp. 337–338, ACM, New York, NY, USA (1998).

[63] D. Ponceleon, S. Srinivasan, A. Amir, D. Petkovic, and D. Diklic, "Key to effective video retrieval: effective cataloging and browsing," in *Proc. Sixth ACM Intl. Conf. Multimedia*, pp. 99–107, ACM, New York, NY, USA (1998).

[64] S. Uchihashi, J. Foote, A. Girgensohn, and J. Boreczky, "Video Manga: generating semantically meaningful video summaries," in *Proc. Seventh ACM Intl. Conf. Multimedia (Part 1)*, pp. 383–392, ACM Press, New York, NY, USA (1999).

[65] S. Sull, J. Kim, Y. Kim, H. Chang, and S. Lee, "Scalable hierarchical video summary and search," *Proc. SPIE* **4315**, 553–562 (2001).

[66] G. Geisler, G. Marchionini, B. Wildemuth, A. Hughes, M. Yang, T. Wilkens, and R. Spinks, "Video browsing interfaces for the open video project," in *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, pp. 514–515, ACM, New York, NY, USA (2002).

[67] J. Graham and J. Hull, "A paper-based interface for video browsing and retrieval," in *Proc. 2003 Intl. Conf. Multimedia and Expo*, **2**, pp. II - 749–52 (2003).

[68] M. G. Christel, "Supporting video library exploratory search: when storyboards are not enough," in *CIVR '08: Proc. 2008 Intl. Conf. Content-based Image and Video Retrieval*, pp. 447–456, ACM, New York, NY, USA (2008).

[69] G. Geisler, "The open video project: redesigning a digital video digital library," presented at the American Society for Information Science and Technology Information Architecture Summit, Austin, Texas (2004).

[70] Y. Deng and B. S. Manjunath, "Content-based search of video using color, texture, and motion," in *Proc. Intl. Conf. Image Processing*, **2**, 534–537, IEEE (1997).

[71] T. Tse, G. Marchionini, W. Ding, L. Slaughter, and A. Komlodi, "Dynamic key frame presentation techniques for augmenting video browsing," in *AVI '98: Proc. Working Conf. Advanced Visual Interfaces*, pp. 185–194, ACM, New York, NY, USA (1998).

[72] R. Hammoud, *Interactive Video: Algorithms and Technologies (Signals and Communication Technology)*, Springer-Verlag, New York, Secaucus, NJ, USA (2006).

[73] S. Srinivasan, D. Ponceleon, A. Amir, and D. Petkovic, "What is in that video anyway?: In search of better browsing," in *Proc. IEEE Intl. Conf. Multimedia and Expo*, pp. 388–392 (2000).

[74] D. Heesch, P. Howarth, J. Magalhães, A. May, M. Pickering, A. Yavlinsky, and S. Ruger, "Video retrieval using search and browsing," in *TREC Video Retrieval Evaluation Online Proc.* (2004).

[75] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and TRECVid," in *MIR '06: Proc. 8th ACM Intl. Workshop Multimedia Information Retrieval*, pp. 321–330, ACM Press, New York, NY, USA (2006).

[76] A. Ghoshal, S. Khudanpur, J. Magalhães, S. Overell, and S. Rüger, "Imperial College and Johns Hopkins University at TRECVID," in *TRECVid 2006 – Text Retrieval Conference, TRECVID Workshop*, 13-14 November 2006, Gaithersburg, Maryland (2006).

[77] M. Rautiainen, M. Varanka, et al., "TRECVID 2005 Experiments at MediaTeam Oulu," in *TRECVid 2005* (2005).

[78] M. Rautiainen and T. Ojala, "Cluster-temporal browsing of large news video databases," in *IEEE International Conf. Multimedia and Expo* (2004).

[79] M. Campbell, A. Haubold, S. Ebadollahi, M. R. Naphade, A. Natsev, J. Seidl, J. R. Smith, J. Tešić, and L. Xie, "IBM Research TRECVID-2006 Video Retrieval System," in *TRECVID 2006 – Text Retrieval Conference, TRECVID Workshop*, November 2006, Gaithersburg, Maryland (2006).

[80] W. Bailer, C. Schober, and G. Thallinger, "Video content browsing based on iterative feature clustering for rushes exploitation," in *TRECVID 2006 – Text Retrieval Conference, TRECVID Workshop*, November 2006, Gaithersburg, Maryland (2006).

[81] C. Foley, C. Gurrin, G. Jones, C. Gurrin, G. Jones, H. Lee, S. McGivney, N. E. O'Connor, S. Sav, A. F. Smeaton, and P. Wilkins, "TRECVid 2005 Experiments at Dublin City University," in *TRECVid 2005 – Text Retrieval Conference, TRECVID Workshop*, 14-15 November 2005, Gaithersburg, Maryland (2005).

[82] P. Dietz and D. Leigh, "DiamondTouch: a multi-user touch technology," in *UIST '01: Proc. 14th Annual ACM Symp. User Interface Software and Technology*, pp. 219–226, ACM Press, New York, NY, USA (2001).

[83] O. Holthe and L. Ronningen, "Video browsing techniques for web interfaces," in *3rd IEEE Consumer Communications and Networking Conference, 2006*, **2**, 1224–1228 (2006).

[84] R. Villa, N. Gildea, and J. Jose, "FacetBrowser: a user interface for complex search tasks," in *Proc. 16th Annual ACM International Conference on Multimedia 2008*, pp. 489–498, ACM Press, Vancouver, British Columbia, Canada (2008).

[85] F. Hopfgartner, T. Urruty, D. Hannah, D. Elliott, and J. M. Jose, "Aspect-based video browsing – a user study," in *ICME'09 - IEEE Intl. Conf. on Multimedia and Expo*, pp. 946–949, IEEE, New York, USA (2009).

[86] M. Halvey, D. Vallet, D. Hannah, and J. M. Jose, "Vigor: a grouping oriented interface for search and retrieval in video libraries," in *JCDL '09: Proc. 9th ACM/IEEE-CS Joint Conf. Digital Libraries*, pp. 87–96, ACM, New York, NY, USA (2009).

[87] J. Adcock, M. Cooper, and J. Pickens, "Experiments in interactive video search by addition and subtraction," in *Proc. 2008 Intl. Conf. Content-based Image and Video Retrieval*, pp. 465–474, ACM, New York, NY, USA (2008).

[88] S.-Y. Neo, H. Luan, Y. Zheng, H.-K. Goh, and T.-S. Chua, "Visiongo: bridging users and multimedia video retrieval," in *CIVR '08: Proc. 2008 Intl. Conf. Content-based Image and Video Retrieval*, pp. 559–560, ACM, New York, NY, USA (2008).

[89] J. S. Boreczky and L. A. Rowe, "Comparison of video shot boundary detection techniques," in *Proc. SPIE* **2670**, 170–179 (1996).

[90] H. Lee, A. F. Smeaton, N. E. O'Connor, and B. Smyth, "User evaluation of Físchlár-News: An automatic broadcast news delivery system," *ACM Trans. Inf. Syst.* **24**(2), 145–189 (2006).

[91] M. J. Pickering, L. W. C. Wong, and S. M. Rüger, "Anses: Summarisation of news video," in *Conf. Image Video Retrieval*, E. M. Bakker, T. S. Huang, M. S. Lew, N. Sebe, and X. S. Zhou, Eds., *Lecture Notes in Computer Science* **2728**, 425–434, Springer (2003).

[92] F. Hopfgartner, D. Hannah, N. Gildea, and J. M. Jose, "Capturing multiple interests in news video retrieval by incorporating the ostensive model," in *PersDB'08 - Second Intl. Workshop on Personalized Access, Profile Management, and Context Awareness in Databases*, Auckland, New Zealand, pp. 48–55, VLDB Endowment (2008).

[93] F. Hopfgartner and J. M. Jose, "Semantic user modelling for personal news video retrieval," in *MMM'10 - 16th Intl. Conf. Multimedia Modeling*, Chongqing, China, Springer Verlag, **1**, 336–349 (2010).

[94] B. M. Wildemuth, G. Marchionini, M. Yang, G. Geisler, T. Wilkens, A. Hughes, and R. Gruss, "How fast is too fast?: evaluating fast forward surrogates for digital video," in *JCDL '03: Proc. 3rd ACM/IEEE-CS Joint Conf. Digital Libraries*, pp. 221–230, IEEE Computer Society, Washington, DC, USA (2003).

[95] A. Divakaran, C. Forlines, T. Lanning, S. Shipman, and K. Wittenburg, "Augmenting fast-forward and rewind for personal digital video recorders," in *IEEE Intl. Conf. Consumer Electronics (ICCE), Digest of Technical Papers*, pp. 43–44 (2005).

[96] K. Wittenburg, C. Forlines, T. Lanning, A. Esenther, S. Harada, and T. Miyachi, "Rapid serial visual presentation techniques for consumer digital video devices," in *Proc. 16th Annual ACM Symp. User Interface Software and Technology*, pp. 115–124, ACM, New York, NY, USA (2003).

[97] S. Shipman, A. Divakaran, M. Flynn, and A. Batra, "Temporal-Context-Based Video Browsing Interface for PVR-Enabled High-Definition Television Systems," *Intl. Conf. Consumer Electronics, Technical Digest*, pp. 353–354 (2006).

[98] M. Campanella, R. Leonardi, and P. Migliorati, "The Future-Viewer visual environment for semantic characterization of video sequences," in *Proc. 2005 Intl. Conf. Image Processing*, 11–14 September 2005, Genoa, Italy, **1**, 1209–1212, IEEE (2005).

[99] M. Campanella, R. Leonardi, and P. Migliorati, "An intuitive graphic environment for navigation and classification of multimedia documents," in *Proc. 2005 IEEE Intl. Conf. Multimedia and Expo*, 6–9 July 2005, Amsterdam, The Netherlands, pp. 743–746, IEEE (2005).

[100] A. Axelrod, Y. Caspi, A. Gamliel, and Y. Matsushita, "Interactive video exploration using pose slices," in *Intl. Conf. Computer Graphics and Interactive Techniques*, ACM Press, New York, NY, USA (2006).

[101] A. Hauptmann, W. Lin, R. Yan, J. Yang, and M. Chen, "Extreme video retrieval: joint maximization of human and computer performance," in *Proc. 14th Annual ACM Intl. Conf. Multimedia*, pp. 385–394, ACM Press, New York, NY, USA (2006).

[102] H. Eidenberger, "A video browsing application based on visual MPEG-7 descriptors and self-organising maps," *Intl. J. Fuzzy Systems* **6**(3), 125–138 (2004).

[103] T. Bärecke, E. Kijak, A. Nurnberger, and M. Detyniecki, "VideoSOM: A SOM-based interface for video browsing," *Lecture Notes ln Computer Science* **4071**, 506 (2006).

[104] H. Goeau, J. Thievre, M. Viaud, and D. Pellerin, "Interactive visualization tool with graphic table of video contents," in *2007 IEEE Intl. Conf. Multimedia and Expo*, pp. 807–810 (2007).

[105] O. de Rooij, C. Snoek, and M. Worring, "Query on demand video browsing," in *Proc. 15th Intl. Conf. Multimedia*, pp. 811–814, ACM Press, New York, NY, USA (2007).

[106] C. Snoek, I. Everts, J. van Gemert, J. Geusebroek, B. Huurnink, D. Koelma, M. van Liempt, O. de Rooij, K. van de Sande, A. Smeulders, et al., "The MediaMill TRECVid 2007 semantic video search engine," *TREC Video Retrieval Evaluation Online Proc.* (2007).

[107] B. Adams, S. Greenhill, and S. Venkatesh, "Temporal semantic compression for video browsing," in *Proc. 13th Intl. Conf. Intelligent User Interfaces*, pp. 293–296, ACM, New York, NY, USA (2008).

[108] M. Jansen, W. Heeren, and B. van Dijk, "Videotrees: Improving video surrogate presentation using hierarchy," in *Intl. Workshop Content-Based Multimedia Indexing*, pp. 560–567 (2008).

[109] W. Ding, G. Marchionini, and D. Soergel, "Multimodal surrogates for video browsing," in *Proc. Fourth ACM Conf. Digital Libraries*, pp. 85–93, ACM, New York, NY, USA (1999).

[110] A. Goodrum, "Multidimensional scaling of video surrogates," *J. Am. Soc. Information Science* **52**(2), 174–182 (2001).

[111] A. Hughes, T. Wilkens, B. Wildemuth, and G. Marchionini, "Text or pictures? An eye-tracking study of how people view digital video surrogates," *Lecture Notes in Computer Science*, pp. 271–280 (2003).

[112] Y. Song and G. Marchionini, "Effects of audio and visual surrogates for making sense of digital video," in *Proc. SIGCHI Conf. Human Factors in Computing Systems*, p. 876, ACM (2007).

[113] B. Wildemuth, G. Marchionini, T. Wilkens, M. Yang, G. Geisler, B. Fowler, A. Hughes, and X. Mu, "Alternative surrogates for video objects in a digital library: users'

perspectives on their relative usability," *Lecture Notes in Computer Science*, pp. 493–507 (2002).

[114] L. Slaughter, B. Shneiderman, and G. Marchionini, "Comprehension and object recognition capabilities for presentations of simultaneous video key frame surrogates," *Lecture Notes in Computer Science*, pp. 41–54 (1997).

[115] M. Campanella, R. Leonardi, and P. Migliorati, "Interactive visualization of video content and associated description for semantic annotation," *Signal Image Video Processing* **3**(2), 183–196 (2009).

[116] C. G. M. Snoek, J. C. van Gemert, T. Gevers, B. Huurnink, D. C. Koelma, M. van Liempt, O. de Rooij, K. E. A. van de Sande, F. J. Seinstra, A. W. M. Smeulders, A. H. C. Thean, C. J. Veenman, and M. Worring, "The MediaMill TRECVid 2006 semantic video search engine," in *Proc. 4th TRECVid Workshop* (November 2006).

[117] C. G. M. Snoek, K. E. A. van de Sande, O. de Rooij, B. Huurnink, J. C. van Gemert, J. R. R. Uijlings, J. He, X. Li, I. Everts, V. Nedovi, M. van Liempt, R. van Balen, F. Yan, M. A. Tahir, K. Mikolajczyk, J. Kittler, M. de Rijke, J.-M. Geusebroek, T. Gevers, M. Worring, A. W. Smeulders, and D. C. Koelma, "The MediaMill TRECVid 2008 semantic video search engine," in *Proc. 6th TRECVid Workshop* (November 2008).

Klaus Schoeffmann is an assistant professor at the Institute of Information Technology, Klagenfurt University, Austria. His research focuses on collaborative video search and browsing, video summarization, video retrieval, and video content analysis. He received a MSc in applied computer science in 2005 and a PhD in distributed multimedia systems in 2009. He is the author of several refereed international conference and journals papers and a member of the IEEE.

Frank Hopfgartner is a doctoral candidate in information retrieval at the University of Glasgow, Scotland and research associate with the Multimedia & Vision Group at Queen Mary, University of London. He received a Diplom-Informatik (MSc equivalent) degree from the University of Koblenz-Landau, Germany. His research interests include interactive video retrieval with a main focus on relevance feedback and adaptive search systems. He is a member of the British Computer Society (BCS), BCS Information Retrieval Specialist Group and ACM SIGIR.

Oge Marques is an associate professor in the Department of Computer & Electrical Engineering and Computer Science at Florida Atlantic University in Boca Raton, Florida. He received his PhD in computer engineering from Florida Atlantic University in 2001, his MS in electronics engineering from Philips International Institute, Eindhoven, Netherlands, in 1989, and his BS in electrical engineering from Universidade Tecnológica Federal do Paraná (UTFPR), Curitiba, Brazil, in 1987. His research interests have been focused on image processing, analysis, annotation, search, and retrieval; human and computer vision; and video processing and analysis. He has published three books, several book chapters, and more than 40 refereed journal and conference papers in these fields. He is a senior member of the ACM and the IEEE, and a member of the honor societies of Tau Beta Pi, Sigma Xi, Phi Kappa Phi, and Upsilon Pi Epsilon.

Joemon M. Jose is a professor at the Department of Computing Science, University of Glasgow. His research is focused on adaptive and personalized search systems, multimodal interaction for information retrieval, and multimedia mining and search. He has published widely in these areas and leads the Multimedia Information Retrieval group at the University of Glasgow. He holds a PhD in information retrieval, an MS in software systems, and an MSc in statistics. He is a Fellow of BCS and a member of ACM, IEEE, and Institution of Engineering and Technology (IET).

Laszlo Boeszoermenyi has been a full professor of computer science and head of the Department for Information Technology at Klagenfurt University since 1992. He is a senior member of ACM and a member of IEEE and Österreichische Computer Gesellschaft (OCG). His research is currently focused on distributed multimedia systems, with special emphasis on adaptation, video delivery infrastructures, interactive video exploration, and multimedia languages. He is the author of several books, and he publishes regularly in refereed international journals and conference proceedings.