Facilitating Video Access by Visualizing Automatic Analysis

Andreas Girgensohn, John Boreczky, Lynn Wilcox, and Jonathan Foote

FX Palo Alto Laboratory
3400 Hillview Avenue
Palo Alto, CA 94304
{andreasg, johnb, wilcox, foote}@pal.xerox.com

ABSTRACT When reviewing collections of video such as recorded meetings or presentations, users are often interested only in an overview or short segments of these documents. We present techniques that use automatic feature analysis, such as slide detection and applause detection, to help locate the desired video and to navigate to regions of interest within it. We built a web-based interface that graphically presents information about the contents of each video in a collection such as its keyframes and the distribution of a particular feature over time. A media player is tightly integrated with the web interface. It supports navigation within a selected file by visualizing confidence scores for the presence of features and by using them as index points. We conducted a user study to refine the usability of these tools.

KEYWORDS Content-based access, video, multimedia, keyframes, speaker identification, automatic analysis, visualization, skimming.

1. INTRODUCTION

In recent years, we have seen increasing availability and use of digital video. Inexpensive mass storage and efficient data compression have made it possible to store large amounts of video on-line. The ability to watch any full-length video is useful, but for many applications, a user simply wants to skim through one of more long videos to get the gist of their content. Alternately, a user may want to watch a short segment of a particular video.

If videos are transcribed or otherwise labeled with text, text retrieval techniques can be used to locate passages of interest (Christel 1998, Brown 1995). For the types of video we are interested in (e.g., weekly group meetings), the effort of adding that information is often not acceptable. Instead, we use automatic analysis techniques that identify features in the media such as certain sounds in the audio or certain types of shots in the video. These features support access to video even if no textual information is available.

While much research has been done on the automatic analysis of video content (Arman 1994, Hampapur 1997, Zhang 1993), less attention has been directed at how to make such analysis useful to the user. Ideally, automatic analysis can make multimedia data less opaque by giving the user an indication of the contents (e.g., shot change (Zhang 1993) and face detection (Wang 1998)). However, even the most sophisticated analysis will be of little value unless it can be presented in a manner appropriate for the user.

This paper presents techniques and applications for interacting with the results of such analyses to aid content-based location, retrieval, and playback of potentially relevant video data. The details of the analysis techniques used in this paper are discussed in (Foote 1998).

In this paper, we describe interaction techniques for accessing video facilitated by automatic analysis techniques. After describing our video environment, we present the analysis techniques used to automatically create video indices. We then show how this information can augment user interfaces that display available videos and play back selected video files. We conclude with the results of a user study designed to help us refine the useful features of our interfaces, and a discussion of future research.

2. THE PROBLEM

At our company, weekly staff meetings and other seminars and presentations are held in a conference room outfitted with several video cameras and microphones. All formal meetings and most presentations are videotaped, MPEG-encoded, and made available to the staff via the company intranet. These videos amount to about three hours per week; currently we have more than 120 hours of video in our database.

In such an environment, users often want to retrieve information such as “the name of the executive visiting next week” or “the report Jim gave about a conference in Monterey.” Finding the desired few minutes in a one-hour staff meeting can be problematic. If users do not remember at which meeting the desired information was presented, they might have to play through several hours of video to find the desired segment. Other users might have missed a meeting and want to review it without having to spend a whole hour watching the entire video.

We want to help users locate specific video passages quickly and provide them with visual summaries of the videos. We want to accomplish this without manual preparation work such as manual transcription and/or speaker identification. Therefore, we use a number of automatic techniques that provide indices into the video material and interfaces that allow users to browse video using those indices.

3. AUTOMATIC MEDIA ANALYSIS TECHNIQUES

Useful information may be automatically derived from sources such as audio and video. This information, or *metadata*, can be generally described as a time-dependent value or values that are synchronous with the source media. For example, metadata might come from the output of a face-recognition or speaker-identification algorithm.

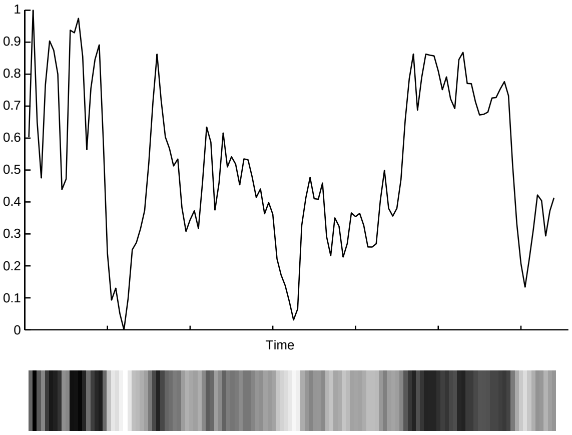


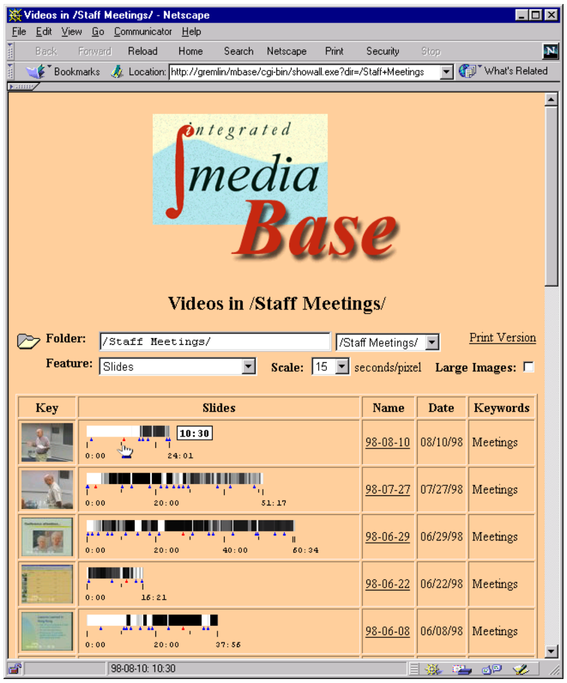
Figure 1: Mapping confidence scores to gray levels

Metadata for video materials can be derived from the analysis of the audio and video streams. For audio, we identify features such as silence, applause, and speaker identity. For video, we find features such as shot boundaries, presentation slides, and close-ups of human faces. These provide several dimensions of data for our browser and player. To detect video features such as presentation slides, we use statistical models applied to discrete cosine transform coefficients that are trained with a set of example slides (Girgensohn 1999b). In our setting, such models recognize more than 85% of all slides while having less than 10% false positives.

Because automatic techniques do not always work reliably, rather than provide a yes or no answer, it is useful to translate metadata values into a “confidence score” for presentation to the user (Foote 1998). For example, rather than having a binary decision for the presence or absence of a feature, we present the user with an interface that shows degree of certainty in decision.

3.1 Using Automatic Analysis

Even the most powerful analysis is useless unless it can be made meaningful to the user. With the analysis techniques described above, the amount of automatically generated metadata can be overwhelming; it is common to generate multiple data points for each video frame, at a rate of 30 per second. An effective method for presenting confidence scores is a graphical visualization, as in Figure 1, in which the confidence score for a feature over time is depicted by levels of gray. Automatic analysis will never be perfect, and will sometimes yield inaccurate metadata. Our approach is to acknowledge that the metadata is inaccurate, but that hiding the errors achieves little. Conversely, presenting it fully as a confidence score lets the user decide what is important. In addition, user-selectable thresholds can take full advantage of all the metadata.


Videos in /Staff Meetings/ - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Location: http://gremlin/mbase/cgi-bin/showall.exe?dir=/Staff+Meetings
What's Related
integrated media Base

Videos in /Staff Meetings/

Folder: /Staff Meetings/ /Staff Meetings/ Print Version

Feature: Slides Scale: 15 seconds/pixel Large Images: □

Key	Slides	Name	Date	Keywords
	10:30 0:00 24:01	98-08-10	08/10/98	Meetings
	0:00 20:00 51:17	98-07-27	07/27/98	Meetings
	0:00 20:00 40:00 60:34	98-06-29	06/29/98	Meetings
	0:00 16:21	98-06-22	06/22/98	Meetings
	0:00 20:00 37:56	98-06-08	06/08/98	Meetings

98-08-10: 10:30

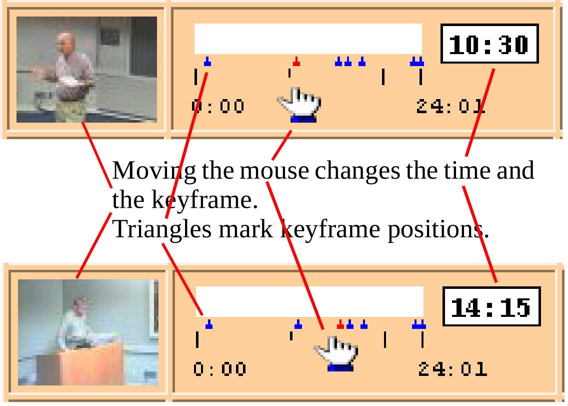
Figure 2: Web-based video directory browser

4. WEB-BASED VIDEO DIRECTORY BROWSER

To provide access to our video collection, we implemented a web-based browser that presents directory listings of videos (see Figure 2). The directories organize videos by content (e.g., staff meetings or conference reports) and sort them by date within each directory. Clicking on a video opens a viewer to play it. The use of a web browser and the MPEG file format enables casual access to the video archive for almost all potential users without the need for additional software or plug-ins.

4.1 Keyframe Use

We enhance each video directory listing with representative frames to help recognize the desired video, and to provide access points into the video. Well-chosen keyframes can help video selection and make the listing more visually appealing. Because it is both difficult to determine a single frame that best represents the whole video as well as to distinguish videos based on a single keyframe, we provide a number of keyframes. The positions of the keyframes are marked by blue triangles along a mouse-sensitive time scale adjacent to the keyframe (see Figure 3). As the mouse moves over the time scale, the keyframe for the corresponding time is shown and the triangle for that keyframe turns red. This method shows only a single keyframe at a time, preserving screen space while making other frames accessible through simple mouse motion. This interface supports very quick skimming that provides a good impression of the content of the video. Clicking anywhere on the time scale opens the video and starts playback at the corresponding time (by invoking the Metadata Media Player described in the next section). Using multiple keyframes in this way gives users an idea of the context and temporal structure of a video.



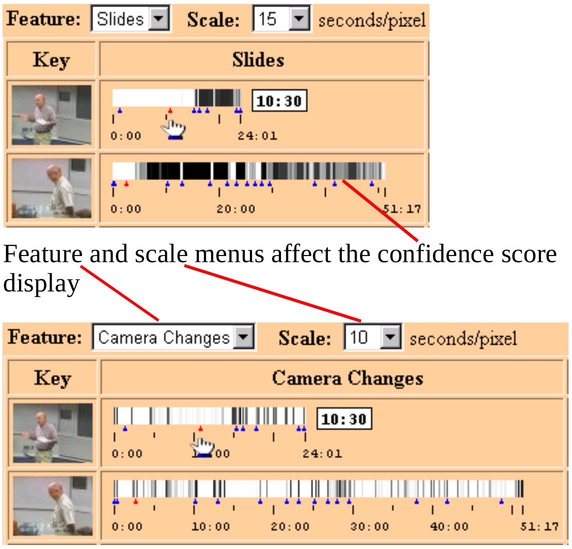
Moving the mouse changes the time and the keyframe.
Triangles mark keyframe positions.

Figure 3: Keyframes attached to time scale

We experimented with animating keyframes by constantly rotating through the keyframes sequence associated with a video, but a display with dozens of animated images proved to be very distracting, even when image changes were synchronized to each other. Access to keyframes via mouse movements gives a better impression of temporal sequence because it is correlated with mouse position.

We automatically determine a number of keyframes relative to the length of the video as discussed in (Girgensohn 1999a). Keyframe extraction is independent of any confidence score computed by other automatic analysis techniques. We found that 25 keyframes per hour of video work well for our default display. These keyframes are not distributed evenly over the length of the video but are concentrated in areas of interest. Our approach differs from other keyframe extraction methods (e.g., (Christel 1998, Zhang 1995)) in that it can determine an arbitrary number of non-evenly spaced keyframes that does not depend on the number of shots in the video.

We determine keyframes for an entire video by clustering video frames using a distance measure based on color histograms (Zhang 1993). This yields clusters that match human perception of similarity in most cases. In addition, temporal constraints for keyframe distribution and spacing are applied. Our approach produces keyframes that summarize a video and provide entry points to areas of interest. We store the clusters produced by the hierarchical clustering of the video frames so that any number of keyframes can be determined rapidly at presentation



Feature: Slides Scale: 15 seconds/pixel

Feature: Camera Changes Scale: 10 seconds/pixel

Feature and scale menus affect the confidence score display

Figure 4: Confidence score display

time. In this fashion, additional detail can be presented on demand. A static display that shows multiple keyframes for each video is available for printing.

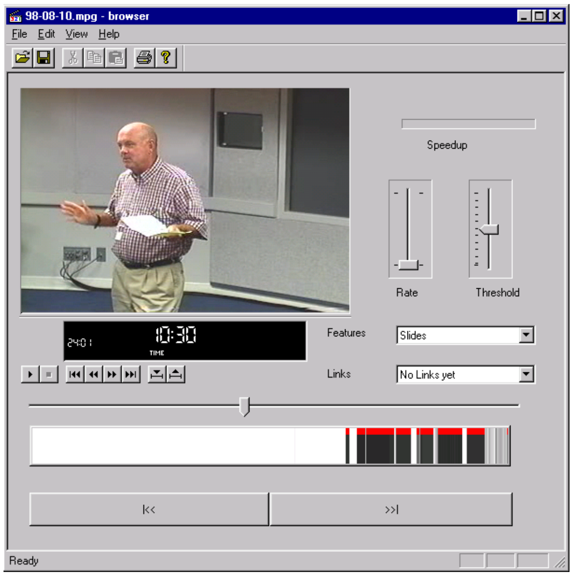
4.2 Feature Display

Different features such as camera changes and slides can be selected from the pull-down menu above the video listing in the web browser. To show how a feature varies with time, it is represented graphically in the time scale such that the shade of gray indicates the confidence level (see Figure 4). High confidence areas are marked in black while areas of lower confidence fade progressively to white, which indicates minimum confidence. Different features can be selected from a pull-down menu. For example, the display of the confidence for presentation slides provides a quick indication for the meetings with slide presentations as well as entry points to those presentations.

The feature time scale can be shown at different resolutions to support zooming in and out (see the "Scale" menu in Figure 4). Presenting the feature graphically aids users in selecting the appropriate video. For example, if they seek a long presentation from a particular speaker, they can ignore videos containing only short examples of that speaker. When launching the Metadata Media Player for viewing a video, the presented feature is automatically selected in the Metadata Media Player, so that locating high confidence areas is rapid and easy.

5. METADATA MEDIA PLAYER

After finding one or more videos in the directory listing, the user must still investigate the videos to find the relevant one(s). It is not simple to determine



98-08-10.mpg - browser

Features: Slides

Links: No Links yet

Ready

Figure 5: Metadata Media Player

whether a long video contains desired information without watching it in its entirety. Standard MPEG players only provide access at the granularity of whole videos. We developed a Metadata Media Player that allows finer-grained access by taking advantage of the metadata extracted from the video. While there are convenient methods for the graphical browsing of text, e.g., scroll bars, "page-forward" commands, and word-search functions, existing video playback interfaces almost universally adopt the "VCR" metaphor. To scan an entire video, it must be auditioned from start to finish to ensure that no parts are missed. Even if there is a "fast forward" button or a slider for scrubbing,[1] it is generally a hit-or-miss operation to find a desired section in a lengthy video. The Metadata Media Player represents a dynamic time-varying process (video) by a static display that can be taken in at a glance.

Figure 5 shows the player interface. The usual transport controls are placed just below the video window. To the right of the window is a menu that selects which confidence score to display. In our case, features are "slides," "camera changes," and "applause&laughter." Confidence scores are displayed time-synchronously below the video slider. The confidence score gives valuable cues to interesting regions in the source stream by using the time axis for random-access into the source media stream. For example, from Figure 5 it is obvious that slides were presented in the last third of the meeting but not at the beginning. Selecting a point or region on the time axis starts media playback from the corresponding time. Clicking at the start of the initial dark bar

[1] Scrubbing is moving the thumb of the time slider slowly so that video images are displayed in sequence.

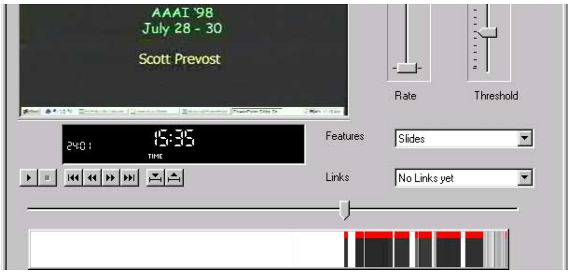


Figure 6: Skipping to the first slide

will start playback with the first slide (see Figure 6). Thus time intervals of high potential interest can be visually identified from the confidence display and easily reviewed without a linear search.

Another way to use confidence scores is to threshold them, that is, to find the times when the confidence score is above, below, or crossing a certain threshold value. The threshold control (above the feature menu) determines index points by thresholding the selected confidence score (see Figure 7). Interface buttons change the current playback time to the next (or previous) index point. Unlike many other implementations, the threshold is user-selectable: a high threshold yields fewer index points and thus a coarser time granularity, while a lower threshold allows finer placement. This is helpful for several reasons: in an area of large confidence variation (many index points), the user can select the most significant indication by increasing the threshold. In a region of small confidence scores the user can still find index points within the region by reducing the threshold, though they may be less reliable. At the bottom of Figure 5 the buttons labeled "|<<" and ">>|" move the playback point to the previous or next index point, as determined from the threshold. The index points are marked in red at the top of the confidence score display.

6. USER STUDY

An initial version of the video database system was deployed at our company in the spring of 1998. It consisted of a web-based directory browser that presented for each video the title, the date, the duration, and a list of keywords. Clicking on the title of a video opened the Microsoft Media Player and started playing the video from the beginning.

In order to investigate the usage of the system, we conducted a survey of 13 employees at our company. Users described how often they used the current system, which features they liked, and which features they thought were missing. The survey pointed out several features that would make the system more useful and increase usage.

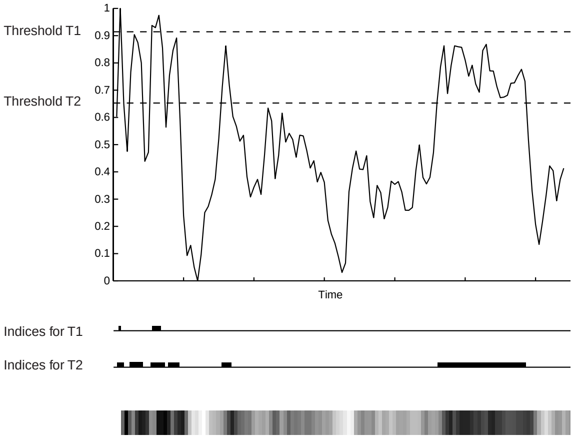


Figure 7: Index points for different thresholds

The system was modified to include a Java applet to display multiple keyframes for each video and a Metadata Media Player that displayed confidence score information. This modified system was deployed to a small group of video researchers at our company.

Early feedback led us to believe that we had included a useful set of features, but that the interface might be too difficult for novices to use. We decided to conduct a small study to observe user behavior during a set of typical browsing tasks using our initial design and the modified design.

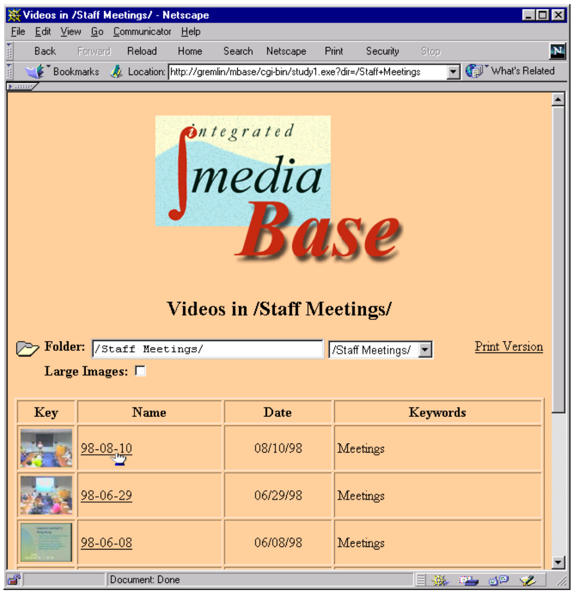
6.1 Participants

Twelve participants (8 male, 4 female) were used in the study. The participants were a mix of researchers, support staff, and administrative staff with varying degrees of expertise in using video browsing software.

6.2 Materials

We created two versions of each interface. One set consisted of a simple web-based directory listing that resembled our initial release (see Figure 8) coupled with a simple media player that contained the standard ActiveMovie controls (see Figure 9). The second set consisted of our latest release version of the web-based directory browser and the Metadata Media Player described earlier in this paper and shown in Figures 2 and 5. All of the applications were instrumented to record the time of every mouse action to the nearest millisecond.

We created six information retrieval tasks, labeled A through F, that were representative of the typical activities of our users. Tasks A, C, and D required finding information that was presented on slides shown during a trip report. The information was shown for 3.3 seconds (D), 10.4 seconds (A), and 324 seconds (C), respectively. Task B required find-


Videos in /Staff Meetings/ - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Location: http://gremlin/mbase/cgi-bin/study1.exe?dir=/Staff+Meetings
What's Related
integrated media Base
Document: Done

Videos in /Staff Meetings/

Folder: /Staff Meetings/ /Staff Meetings/ Print Version

Large Images: □

Key	Name	Date	Keywords
	98-08-10	08/10/98	Meetings
	98-06-29	06/29/98	Meetings
	98-06-08	06/08/98	Meetings

Figure 8: Video directory browser without metadata

ing information presented for 1.3 seconds in the audio by a specified speaker, but the task description mentioned several visual cues that occurred within 110 seconds before the audio segment. Task E required counting the number of award recipients which were shown in a 5-minute video segment. Task F (which was performed last by participants) required finding the video that contained a particular trip report. This could be verified from information on several slides that were shown for 7 minutes.

The six relevant staff meeting videos were presented in the directory browser, and no non-relevant videos were included in the list. Our automatic analysis techniques were applied to the videos to create three feature indices: 1) camera changes, 2) detected slides, and 3) laughter and applause.

6.3 Procedure

The experiment used a between-subjects design. Six randomly chosen participants used the old browser and player, while the other six participants used the new browser and player.

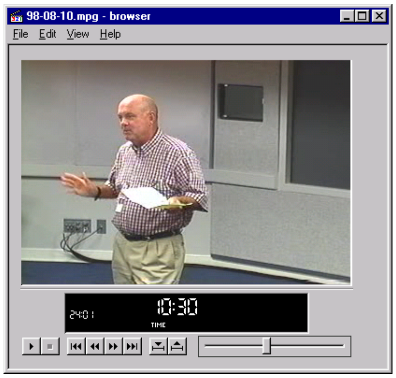


Figure 9: Simple Media Player

Each participant answered an initial survey to discover familiarity with video browsing in general and our system in particular. After the survey, each participant was given a short training session on the browser and player they would use to complete their tasks. This training took one minute for the old system and three minutes for the new system for most participants. Participants were encouraged to use the system until they felt comfortable with the features.

For each task, participants had to read a description of the information they were looking for, find the relevant video segment, and then write the answer on a sheet of paper. All mouse actions were recorded and time-stamped. The participants were videotaped and an experimenter noted the strategy used to discover the information required for each task. The completion time for each task was measured from the first mouse action to the last significant event (either stopping video playback or closing the browser).

An exit survey was used to assess the difficulty of the task and the usefulness of various features. Participants rated interface features on a 5 point scale.

6.4 Results

Participants used a wide variety of strategies to find the required information in the video documents. This led to a large variation in task completion times. There were no significant differences in performance between the two groups ($F_{(1,10)}=1.15$, $p>.3$). The average time to complete each task ranged from 32 seconds to 148 seconds among the participants.

Participants using the new interface commented that the ability to easily see multiple keyframes was useful. They also gave high scores for the usefulness of the display of the confidence scores for locating video passages. Participants using the old interface commented on the need for better information about the contents of the videos. For the new interface group, there were large differences between participants in the rating of interface features, reflecting the use of those features during the experiment.

6.5 Discussion

Participants using the old interfaces had two possible strategies: scrubbing or playing. Participants scrubbed until they saw an image related to the query and then used a combination of playing, scrubbing, and random positioning near the relevant passage until the desired information was found.

Participants using the new interfaces had a wider range of options. Some participants ignored the new features and followed the strategies mentioned above. Most participants moved the mouse along the timeline for the desired video until a keyframe related to the task was shown and then they clicked to start playback at that point. Participants were often confused by the fact that playback started at the point they clicked on the timeline and not at the time of the currently displayed keyframe. Only one participant selected different indices in the web-based directory browser, but all of the participants in this group selected different indices in the player to help them find relevant passages. Some participants used the buttons to jump the next or previous index points, but more often they clicked at the start of the high-confidence areas on the timeline.

For both groups of participants, the first action taken after opening the media player was to stop video playback, usually accompanied by a mild exclamation of displeasure. Autoplay is a feature that works well for users who want to watch a full video from the start, but it aggravated users who wanted to browse. Another problem that both groups encountered was the lack of resolution of the slider and timebar.

Participants using the new interfaces almost always used the indices and other features to help them find the required information quickly and easily. However, the large number of user interface elements occasionally sidetracked the participants, so that their overall task completion times were not significantly faster.

The wide variety of strategies used to complete the given tasks implies that even if we make a simplified interface the default, users should have the ability to activate additional features for specific tasks.

We have made a number of changes to the interfaces based on the results of the user study. The Metadata Media Player no longer starts playing the video upon start-up by default. We plan to study different solutions to the problem of the keyframe image not corresponding with the video playback starting point. The pause button has been eliminated, since it was functionally equivalent to the stop button. We are adding a default index feature that combines the applause and laughter detection feature with slide changes to give a generic “interesting event” feature. We are also adding a zoom feature that expands a small segment of the video to fill the entire timeline, to facilitate fine positioning.

7. RELATED WORK

Previous related work has been concerned with techniques for presenting and browsing continuous media database query results, and with techniques for presenting confidence values for full-text search. Bronson (1992) describes using time-stamped keyframes and keywords to access portions of a video sequence. Yeung *et al.* (1995) cluster keyframes to represent of the structure of a video sequence. Arman *et al.* (1994) uses keyframes supplemented with pixels from the video sequence to represent content and motion within a sequence. In these systems, the keyframes are static and represent fixed points within the video sequence.

Wilcox *et al.* (1994) developed a system for graphically displaying the results of speaker segmentation and classification. The user is presented with the best classification estimate instead of the confidence values of those estimates. Brown *et al.* (1995) provide a set of confidence values for segments of (text) captions. In that system, a set of confidence values can be selected to start playback of the associated video segment. Confidence values from multiple features are not used. Tonomura *et al.* (1993) describe an interface that allows users to visualize multiple index features for video clips. Low-level features are displayed, making comparison difficult, but by mapping to a single timeline, correlations become evident. Their interface is intended as a visualization of the video indices, and is used for content analysis more than for browsing and playback. The STREAMS system (Cruz 1994) supports the browsing of presentation recorded with several cameras. While it does not use automatic analysis techniques, it presents hand-annotated speaker information in a color-coded timeline. None of these systems closely integrates automatic analysis with video content browsing and playback. Many of the systems also rely heavily on text extracted from close captions.

The NewsComm system (Roy 1996) is a hardware device that allows users to navigate stored audio recordings. Automatic analysis is used to generate index points based on speaker changes and speech pauses. Users can skim the audio and skip to the next or previous index point. An early design of this system incorporated a graphical display of the index points available in the audio stream. This display was removed during the design process to simplify the interface, but not directly as a result of user tests.

There are many commercial and research systems used for video database access. The Virage Video Engine (Hampapur 1997) is an example of a typical video indexing and retrieval system. This system provides several different indices, but a limited user interface. Video clips are found by specifying a set of low-level query terms. The metadata that leads to the retrieval of specific clips is not presented to the user. This supports the goal of reducing the information presented to users, but in our opinion, the presentation of confidence scores derived from the metadata provides more support for locating a desired clip quickly and easily for some users.

8. CONCLUSIONS

As video databases become larger and more common, intelligent video browsers will become critical for navigating, locating and accessing multimedia data. In this paper, we describe an interface that presents results from automatic video analysis and a media player for viewing and skimming a particular video. The browser presents a video directory listing that allow users to find a single video clip from a collection of videos. Keyframes appear in the video directory to distinguish different videos. Different metadata types can be selected and a confidence-score scale indicate the likelihood and location of metadata features in the video. Clicking at a point on the confidence scale provides direct playback of the video from the selected point. Both the directory listing and the media player use automatically-generated confidence scores to distinguish videos and to navigate within a single video.

A user study showed that both the dynamic keyframes and the feature scores helped users locate passages of interest. The study also uncovered a number of usability problems that prevented the participants of the study from being more efficient than the members of the control group. We will use the insights gained in the study to improve the user interfaces.

The technique for determining keyframes, the slider interface for changing keyframes, and the display of confidence scores for the presence of features are all innovative elements for facilitating access to video. Use both inside our company and in the user study show that our approach is a promising one. In the future, we plan to address the usability problems uncovered in the study, to introduce new techniques for navigating video, and to conduct a follow-up user study.

REFERENCES

Arman, F., Depommier, R., Hsu A., and Chiu M.-Y. (1994). “Content-based Browsing of Video Sequences,” In *Proc. ACM Multimedia 94*, pp. 97-103.

Bronson, B.S. (1992). “Method and apparatus for indexing and retrieving audio-video data.” US Patent 5,136,655.

Brown, M.G., Foote, J.T., Jones, G.J.F., Spärck Jones, K., and Young, S.J. (1995). “Automatic Content-Based Retrieval of Broadcast News.” In *Proc. ACM Multimedia 95*, pp. 35-43.

Christel, M.G., Smith, M.A., Taylor, C.R., and Winkler, D.B. (1998). “Evolving Video Skims into Useful Multimedia Abstractions,” in *Human Factors in Computing Systems, ACM CHI 98 Conference Proceedings* (Los Angeles, CA), pp. 171-178.

Cruz, G. and Hill, R. (1994). “Capturing and Playing Multimedia Events with STREAMS,” in *ACM Multimedia 94 Proceedings*, New York: ACM Press, pp. 193-200.

Foote, J., Boreczky, J., Girgensohn, A., and Wilcox, L. (1998). “An Intelligent Media Browser using Automatic Multimodal Analysis,” in *ACM Multimedia ’98*, Bristol, England, pp. 375-380.

Girgensohn, A. and Boreczky, J. (1999a). “Time-Constrained Keyframe Selection Technique,” to appear in *ICMCS’99*.

Girgensohn, A. and Foote, J. (1999b). “Video Classification Using Transform Coefficients,” in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing* (Phoenix, AZ), vol. 6, pp. 3045-3048.

Hampapur, A., Gupta, A., Horowitz, B., Shu, C.-F., Fuller, C., Bach, J., Gorkani, M., and Jain, R. (1997). “Virage Video Engine,” in *Storage and Retrieval for Still Image and Video Databases V, Proc. SPIE 3022* (San Jose, CA), pp. 188-197.

Roy, D.K. and Schmandt, C. (1996). “NewsComm: A Hand-Held Interface for Interactive Access to Structured Audio,” in *Human Factors in Computing Systems, CHI 96 Conference Proceedings* (Vancouver, BC), pp. 173-180.

Tonomura, Y., Akutsu, A., Otsuji, K., and Sadakata, T. (1993). “VideoMAP and VideoSpaceIcon: Tools for Anatomizing Video Content,” In *Proc. ACM INTERCHI ‘93*, pp. 131-141.

Wang, C. and Brandstein, M.S. (1998). “A Hybrid Real-time Face Tracking System,” In *Proc. ICASSP 98*, Vol. VI, pp. 3737-3740.

Wilcox, L., Chen, F., and Balasubramanian, V. (1994). “Segmentation of speech using speaker identification.” In *Proc. ICASSP 94*, Vol. S1, pp. 161-164.

Yeung, M.M., Yeo, B.L., Wolf, W. and Liu, B. (1995). “Video Browsing using Clustering and Scene Transitions on Compressed Sequences”, in *SPIE Vol. 2417 Multimedia Computing and Networking 1995*, pp. 399-413.

Zhang, H.J., Kankanhalli, A. and Smoliar, S.W. (1993). “Automatic Partitioning of Full-motion Video,” *Multimedia Systems*, Vol. 1, No. 1, pp. 10-28.

Zhang, H.J., Low, C.Y., Smoliar, S.W., and Wu, J.H. (1995). “Video Parsing, Retrieval and Browsing: An Integrated and Content-Based Solution,” ACM Multimedia 95, pp. 15-24.