TOWARDS A VIDEO BROWSER FOR THE DIGITAL NATIVE

Brett Adams,[†] *Stewart Greenhill,*[†] *Svetha Venkatesh*[‡]

[†]Institute for Multi-sensor Processing and Content Analysis (IMPCA)
Curtin University
Perth, W. Australia

[‡]Centre for Pattern Recognition and Data Analytics (PRaDA)
Deakin University
Geelong, Victoria

ABSTRACT

Almost every aspect of how we create, transmit, and consume video has changed, but video interfaces still mimic those from video's inception. We extend Temporal Semantic Compression for interactive video browsing, which uses an arbitrary frame-by-frame interest measure to sub-sample video in real time, with user interface elements that visualize these measures and the effect of compressing on them. We experiment with a novel interest measure for popularity, and design novel visualizations for expressing interest measures and the compression interaction. We conduct the first formative evaluation of the TSC paradigm, with 8 subjects, and report design implications arising from it.

Fig. 1. Top: Randy Pausch's Last Lecture (cropped), compressed on popularity; Bottom: Youtube *Insight* viewing statistics used for popularity measure. (Courtesy CMU Web Communications)

1. INTRODUCTION

Video, richest of the traditional media, has settled into its home on the web. Video is accruing at a massive rate: Youtube claims 2 billion views a day, and 24 hours of uploads a minute; Cisco estimates that by 2013, 90% of web traffic will be video data. Video comes in increasingly diverse forms: ranging in duration from seconds to hours, it consists of novel genres and meta-genres, such as remixed content, and is consumed in diverse settings, from desktop to couch to train. Thus foraging and filtering have become the common browsing tasks; increasingly, the heaviest browsers are those who have never known life without Youtube.

Consider this scenario: A friend messages you that he has just watched an inspirational video, the late Randy Pausch's Last Lecture–a video of over 12 million views. You hit the video and see that it is over an hour long. You have 5 minutes before you need to be at your origami lesson. Your friend neglected to tell you what inspired him, let alone any timecodes of the relevant portions of video. But all is not lost. You switch on the TSC feature, which enables you to view the Last Lecture through the aggregate appreciation of those 12+ million viewers. You compress the video to 5 minutes, and begin watching at the most popular section, where Randy talks about the people who influence our lives for good.

In earlier work we introduced Temporal Semantic Compression (TSC), a method of subsampling a video interactively according to a frame-by-frame measure of interest [5]–in essence allowing the user to gradually remove or add less interesting sections. Here we demonstrate the use of two variants of a novel interest measure for *popularity*. Moreover, we examine novel ways of communicating the compression gesture, together with interest measures, to the user. We conduct the first formative evaluation of the TSC paradigm, with 8 subjects, and report design implications arising from it. The significance of developing smart interfaces for video, that are nonetheless accessible to a wide demographic, is massive: one need only consult the figures stated above.

2. RELATED WORK

Here we consider work in the class of "information-aided video browsing." Regardless of where information about a video has come from–it could result from automatic content analysis, view statistics, or manual annotation–the question is: how can this information become a useful facet of a browsing interface?

We will focus on work that has some level of interactivity, which rules out the class of video abstractions [15] that produce fixed surrogates [14]. We note, in passing, related work on context-aware scrolling [9], where scrollbars are analogous to video sliders or seek bars, and document analysis is analogous video analysis. Of relevance too is the work on "Scented Widgets" of [16]. Scented widgets are user interface elements that fuse control and visualization of auxilliary information to aid navigation in information spaces. One example given of a scented widget is a slider bar augmented with a bar chart of the number of views of a document. The reader might note the similarity between this and the example, mentioned in Section 3.2, of using per-frame view statistics to drive temporal compression.

Our goal is to create video browsers that take advantage of automatic analysis or their proxies, with highly interactive yet simple interaction methods. This second requirement is necessary to create interfaces that can be used in a casual, non-expert setting, applicable to the range of browsing scenarios that occur in such a broad setting–from goal-driven to serendipitous [10]–where there may be constraints on time or bandwidth. Hence, in considering related work, we will emphasize the aspects of *complexity of interface*, and *degree of interactivity*. Below we examine in some detail exemplars of different design decisions regarding these aspects.

At one extreme, Haubold and Kender [7] present the Video Audio Structure Text MultiMedia (VAST MM) browser for video lecture libraries. As implied by its name, VAST MM is less a browser and more a multimodal search dashboard or workbench aimed at search and retrieval tasks. The browser presents a time-aligned stack of indices for different modes, including keyframes, audio and visual segments, raw text, and topics. Interaction with the displayed summaries is via three sliders: *scene segmentation* affects the granularity of visual segmentation and hence the number of keyframes displayed; *temporal zoom* alters the duration of video represented by one page of summary; and *text context* effects the amount of textual detail displayed. VAST MM has been evaluated on a large library created for Columbia University [6], and with a wealth of use-log and survey data collected over three years. One of the insights arising from their study is that audio-visual browsing–i.e., those that preserve the temporal nature of video–while being slower than key-frame-based summaries, is more effective for goal-driven browsing.

Less complex is the Video Explorer, presented by Schoeffman et al. [13]. The Explorer augments typical video navigation facilities with instruments that visualize low-level, content-based analyses, called interactive navigation summaries (INSs). Summaries include representations for: dominant colour, camera motion, storyboards, and frame stripes (contiguous portions of a frame). Each INS is composed of two diagrams–an overview for the duration of the video, and a detailed view for a smaller, user-defined window. Interactions include enabling/disabling INSs; manipulation of INS size and zoom window duration; and query-by-example using regions of interest. The user can choose to lock the temporal zoom window, thus enforcing synchrony. When all INSs are used in array, they provide a powerful search mechanism, at the cost of some degree of cognitive decoding.

Simpler still, is SmartPlayer, proposed by Cheng et al. [2], another variation on the theme of "adaptive fast-forwarding." It adopts the metaphor of scenic car driving, complete with a dash board depicting a speedometer, and alters the playback speed to be faster during "unexciting" sections and vice versa. The slider bar is shaded to indicate the amount of motion for each point in the video; markers appear beneath the slider indicating "semantic events," which are manually attached, but envisaged as being automatically derived for various genres, e.g., goals in a soccer broadcast. SmartPlayer allows the user to alter the playback speed, and attempts to learn a personalized playback speed with respect to semantic event type by interpolating between the automatically-derived and user-supplied speeds. SmartPlayer mutes audio, and feedback on this aspect was negative.

Hurst [8] proposed a two-dimensional gesture to simultaneously manipulate position and speed for playback, called ZoomSlider. The effect of the gesture is predictable because curvilinear. ZoomSlider does not use content-analysis, but Pongnumkul et al. [12] use a similar gesture for adaptive playback speed that does use content analysis. To achieve a playback speed of, say 10x, their browser selects clips to play at 2x, skips forward approximately 150 frames and plays the shot associated with the nearest automatically selected keyframe. Keyframes are clustered on colour histograms to form a hierarchy for skimming at higher speeds. Because the video speed is coupled to the two-dimensional gesture, the result is a skim, which must be watched linearly.[1] Also, for a given speed, there is no indication of which parts of the video will be included in the skim. In contrast, we decouple "compression level" from other gestures, allowing it to be set and left while subsequent skimming or seeking is performed. The result of a new compression level is displayed on the slider.

Divakaran et al. [3] proposed an adpative, non-linear subsampling approach, based on motion level, which is viewed as an estimate of visual information rate, to adapt playback speed. Subsequent work used visual complexity, which devolved to motion level when spatial complexity is uniform [11]. Details of the interface are not given.

The SmarkSkip interface, Drucker et al. [4], was designed for consumer browsing as an update to fast-forward/rewind. SmartSkip's interface overlays thumbnails of frames before and after the currently viewed frame. The duration between thumbnails is controlled by a zoom factor. SmartSkip was evaluated for two scenarios: commercial skipping, and locating the weather segment in a news broadcast. The authors note the apparent discrepancy between subjective feedback and quantitative results for the experiment tasks. This observation accords with the view that, for the consumer setting, how pleasurable an interface is often has the most influence over a user's willingness to use it.

It can be observed from these interfaces that in order to decrease the apparent complexity of an interface, the details of content-based analysis, or metadata, are hidden and packed

[1]We have experimented with a two-dimensional gesture for simultaneously manipulating compression level and seek, but found it to produce an excessive cognitive load, particularly for the consumer setting.

into a compact visualization, or some kind of non-linear subsampling scheme, or both.

Visualizations, for wide accessibility and adoption, need to be *interpretable*. Some are moreso than others. For example, dominant colour is a direct representation of the represented feature, colour. It is a non-arbitrary mapping, and immediately understandable to anyone who can see colour. Dominant motion mapped to colour, on the other hand, must be decoded with one level of indirection–the colour-to-motion legend–which is arbitrary.

Non-linear sampling likewise needs to be *predictable*. The supreme accessibility of the old, dumb linear subsampling method, such as fast-forward, comes from its predictability. Everyone knows roughly that 8x speed will jump a certain distance from frame to frame. By contrast, the effect of non-linear sampling–whether it be skipping, time-base manipulation, or a combination–often emerges from a convoluted pipeline of algorithms, and is not obvious. The user can be left asking "what just happened?" It is a cardinal rule of interface design (design in general) that a feature's effect be comprehendable, or its state ascertainable. Hidden states, or apparently non-deterministic behaviour, go ill with most users.

The **core concept** of the TSC is *interestingness*. It is represented by a simple, single-dimension curve. Being generic, interestingness can accomodate a wide range of instantiations: action, presence of an actor or object, popularity, mood, etc. The **core interaction** of the TSC is compression on interestingness. Interestigness is generic; and compression is common interaction metaphor. Together they span a wide range of browsing scenarios and interactivity levels. The <same measure + compress> mechanism can yield a static thumbnail (e.g., the most interesting video portion); a video abstract of fixed length (e.g., a movie compressed to 5% of its duration); or a completely interactive browsing mechanism for everything between 0 and 100% of a movie's duration.

3. EXTENDING THE TSC BROWSER

Semantic compression reduces the information load on the user by retaining only important or interesting parts of a video. This requires an interest function that rates the importance of each frame of video. Below we describe an example content-based interest measure, tempo. This explanation is condensed from earlier work [5] as necessary background. Then we enlarge upon the possibilities of interest measures by introducing other, non-content-based examples. Last, we explain the visualizations conceived to represent interest measures to the user in an accessible way, and communicate the effect of the compression interaction.

3.1. Content-based compression

In [5] we used a video *tempo* function, T, derived from low-level motion and audio features. A plot of tempo for the movie Shrek can be seen in the top half of Figure 4.

To compress on this function, the user defines a *compression factor* $0 < f \leq 1$, which controls the amount of information presented. Given f, we compute the duration of the compressed video $\tau = fN$. Given an interest function E, we rank M shots in order of interest, $S_{E,1}, ..., S_{E,M}$. The compressed video is the sequence of shots satisfying $\sum_{i=1}^{k} dur(S_{E,i}) = \tau$, ordered in their original relative position. The value $S_{E,k}$ is defined as the *threshold* value for function E and compresson factor f.

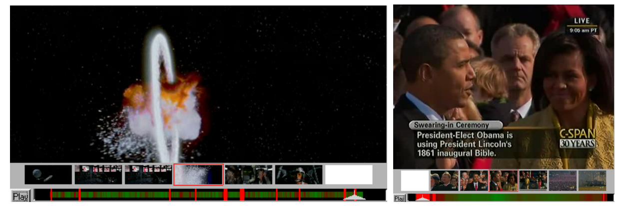


Fig. 2. Left: At high compression on Drama alone, the climax of Starwars remains; Right: Browsing a non-movie by tempo: Barrack Obama's inauguration speech. At high compression, a key event remaining is the swearing-in.

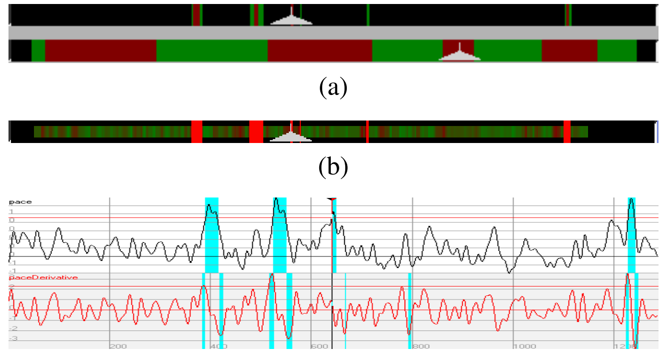
Compressing on T is equivalent to compressing on *action*. We also allow the user to compress on the derivative of tempo, T', because regions of changing tempo often mark important transitions in the narrative. Compressing on T' is equivalent to compressing on *drama*. While feature extraction is potentially time-consuming, the compression function can be calculated almost instantaneously, allowing it to be interactively recomputed during playback. Examples of using tempo to browse can be seen in Figure 2.

3.2. Other measures of interest

In the above section we have demonstrated an example of a content-based interest measure, which is useful for, but not limited to, browsing produced movies. Here we enlarge upon the possibilities of what can constitute an interest measure.

As hinted at earlier, *popularity* can be a useful measure of interest. In Web 2.0 and social media, popularity is a defining characteristic of a site, person, and media item. Popularity can be inferred in many ways. E.g., Carlier et al. infer regions of interest from traces of user panning and zooming to produce video summaries [1]. What is necessary for a TSC browser is an estimate of per-frame or per-shot popularity. One source of time-indexed popularity is viewing statistics, such as those captured by Youtube under their *Insights* metrics. Insights aggregates which sections of a video are viewed in order to calculate an average "hotspots" plot for the video relative to other videos of the same duration (see Figure 1). Another source of time-indexed popularity is manual annotations attached to a video's timeline. Social media application Viddler allows users to tag comments to a video's timeline. Each comment can be interpreted as an index of interesting content at that point, and all comments can be aggregated into to a popularity measure that spans the video's duration. This

Fig. 3. Browser incl. playback, slider and shot thumbnails



(a)

(b)

(c)

Fig. 4. **(a) Discrete** slider in real-time (top) and compressed-time (bottom). Only included shots are displayed, and colour indicates shot class: red for action, and green for drama; **(b) Continuous** slider for real-time. Hue in the colour bar indicates interest level, and included shots are highlighted by height and brightness; **(c)** Tempo and derivative used to derive slider decorations. Shaded regions indicate shots included at current compression.

information is freely available via Viddler's open API.[2] See Figure 5 for an example of browsing by popularity derived from time-indexed comments.

The second example of potential interest measures we will offer here is that of actor or object *presence*. Many browsing scenarios might revolve around finding when a particular person is on screen–e.g., an actor in a movie, or a player in a sports match. This information can be sourced from content-based analysis,[3] from semi-automatic means such as time-indexed references in social media forums or closed captions, or from manual annotation. When presence information is binary, it can be aggregated to yield an interest measure, and when it is in the form of probability values, is even more apt to expression as a continuous interest measure.

3.3. Visualization

This browser is intended to be a drop-in replacement for existing playback widgets. The playback area and slider are augmented with the following:

- The playback slider is decorated with a visual summary of the interest function and included shots.
- The entire widget is resizable. Aspect ratio is maintained, and any additional space is assigned to shot keyframes at and surrounding the current shot.
- Compression is adjusted interactivley during playback with the mouse scroll-wheel. Thumbnails and slider adjust accordingly.

[2]E.g., see Code Rush, a video about the first release of Netscape's source code in 1998, at http://www.viddler.com/explore/waxpancake/videos/12/

[3]Viewdle (www.viewdle.com) purports to offer face recognition on small to medium size databases of good performance in a web service setting.

Playback sliders are used to seek to a particular time relative to the total duration of the video. Here there are two possible time-domains for playback: compressed and real-time. The compressed domain has $\tau = fN$ frames, and contains only included shots; the original real-time domain has N frames. The slider is decorated to indicate interest level for the shots at that position. This is done by assigning each shot an interest level, and projecting this into either the compressed or real-time domain.

Figure 4a. shows two position sliders for the movie Shrek ($\delta = 0.5$, $f = 0.05$). The *discrete* slider shows only the included shots, coloured red for *action* and green for *drama*. When both sliders are displayed, the user can position or playback the movie in either time domain. With higher compression the compressed-time slider gives more precision in positioning, as the included shots are expanded to fill the width of the slider.

Figure 4b. shows the *continuous* slider, where action and drama potential of both included and non-included shots is indicated. Shots with high interest are displayed in red, and shots with low interest are displayed in green. We derive a potential $P \in [0, 1]$ from the interest value, and use this to control the hue. P can be defined in several ways, but this version shows the maximal value of the shot interest $S_{E,i}$ divided by the *threshold* interest value $S_{E,k}$. This assigns all included shots a potential of 1, and excluded shots a potential related to how close they are to the "weakest" included shot. Additionally, included shots are highlighted by varying the brightness and/or height of the colour bar in the slider.

Figure 4c. shows the tempo and its derivative, used to derive the displays above. The shaded regions indicate shots

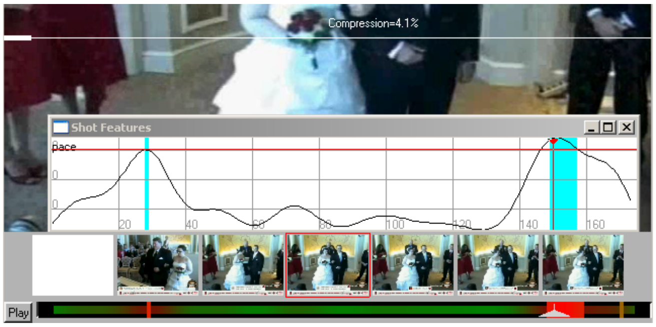


Fig. 5. Browsing by popularity: user-contributed video with community-contributed annotation. (Overlaid is a plot of the underlying interest function, not normally visible, to demonstrate its correspondence with the colour-coded slider)

which are included on the basis of action (top) and drama (bottom). The red horizontal lines indicate the value of the threshold interest level, used to scale the hue for both shots in the continuous slider.

Compared to the discrete slider, the continuous slider shows less information about included shots: it does not indicate shot class (action/drama). But it does show the potential for all shots to be included in the video, allowing the user to identify and seek to areas of interest even for shots not present at a given compression level. This is more suitable when there is only one slider for playback position.

4. EVALUATION

We conducted a formative evaluation of the visualization schemes and compression interactions detailed in Section 3. The aim was to see which interface configurations were the most intuitive, in order to guide subsequent design iterations.

Eight people were recruited from the Department of Computing. This cohort has a bias of general technical proficiency, but reported various degrees of familiarity with video browsing.[4] Participants were given a short oral presentation on the aims of the TSC project–augmenting consumer-level browsing with metadata; were shown how to access the browser on a laptop and the set of movies for use in the evaluation; and were given a sheet containing further instructions and a questionnaire. The instructions invited each participant to spend at least 5 minutes playing with the browser in its various configurations, before attempting two tasks: (i) Browse a familiar movie to find scenes you remember, and (ii) Browse an unfamiliar movie to get a feel for its story or structure.

The available movies included Hollywood feature films: Shrek, The Truman Show, Starwars Episode IV, The Matrix, Raiders of the Lost Ark, Harry Potter and the Philosopher's Stone, Labyrinth; and two videos for browsing by popularity, from social media sources, namely Randy Pausch's Last Lecture[5] and a wedding video.[6]

Users spent on average 15 minutes on the evaluation, which included multiple choice questions, ratings on 5-point Likert scales, and valuable free-response sections. A number of participants also shared their views following the evaluation. Below we report feedback on specific questions.

4.1. Overall

Overall	Undec.	Agree	S. Agree
I like the browser	1	**5**	2
I understand the visualizations	1	**6**	1
I understand the compression interaction	1	**4**	3
I am interested in using it again	2	**5**	1
How helpful was the browser for:	Undec.	Helpful	V. Helpful
Finding scenes in familiar movies	1	**5**	1
Getting a feel for unfamiliar movies	**3**	2	1
Finding popular parts of Youtube/Viddler videos	**4**	2	1

Table 1. Results for overall response and tasks. (Note: No ratings were given below *Undecided*.)

Participants' general response to the browser and task completion are presented in Table 1. In aggregate, they are (embarrassingly) positive; no response was below the mid-point of the 5-point Likert scale used. Responses to within-lab evaluations are known to suffer from bias; and one-off evaluations can suffer from technology placebo effects. These can only be rectified with a larger, longitudinal study. But sampling the free responses gives some insight into the veracity of the positive quantitative feedback.

One participant wrote "Compression feature looks great!" and expressed a desire for interest measures at deeper conceptual levels, such as humour, violence, actor presence, etc. Another participant wanted sports-specific interest functions. This idea was noted in Section 3.2, but was not part of this evaluation. We avoided quantitative measures of task performance, such as time-to-seek, as they are reported in the literature as ill-adapted to casual browsing.

4.2. Visualizing Interest and Compression Effect

Participants' responses to the interest measure visualizations, and the effect of compression, are in Table 2. Discrete colourization–action in red, and drama in green–was preferred, 7 to 1. This was surprising, as the authors believed a continuous mapping of interest measure to colour would indicate useful nuances. On investigation, this preference seems to have stemmed from: (a) the fact that the evaluation material was mostly feature films, and hence the action/drama dichotomy was easily understood; and (b) confusion over the interpretation of the two colours at either pole of the continuous mapping. This confusion stems from the browser's use of gradations between **two** colours to encode **one** concept: interest. Low interest need not be encoded with its own colour.

[4] Average time spent per week browsing videos was between 0–2 hours.

[5] http://www.youtube.com/watch?v=ji5_MqicxSo

[6] http://www.viddler.com/explore/brandice/videos/449/

A simple solution would be to encode interest on a spectrum stretching from a neutral colour (e.g., using the alpha channel) to red. Then "hotspots" would appear as such, and would catch the eye without another colour competing for attention.

The second item in Table 2 is preferences for visualization of operational compression level–i.e., which parts of the video remain in the summary at the current compression level. In Section 3.3 we stated the intended effect of the different visualization techniques is to emphasize, not necessarily contiguous, portions of the timeline. Participants' responses therefore will depend on which they can best see at a glance. Of the options tabulated, the effect that both brightens and heightens included sections received the highest votes at 4 to 1, leaving 3 undecided. This is probably because it emphasizes spatially and with colour, simultaneously. The lone counter vote was due to the participant feeling that the double-emphasis of the alternative left him wondering if some other piece of information were also being encoded at the same time.

The last section of Table 2 records the votes given to different interface elements for the scenario where participants were invited to mix and match elements to create their own custom browser. The feature that received least votes was the slider for compressed domain playback (e.g., the bottom slider in Figure 4a.). There is no doubt that the inclusion of a second slider greatly increases the interface's complexity. One participant suggested that the function of the compressed playback slider be fused with the decorated, normal playback slider. For wide appeal, this is close to a requirement. But it presents a problem. If normal playback and compressed playback are served by one and the same slider, a choice must be made about how these separate functions are fused in a single interface element. One option is to view them as alternate *modes*. This would necessitate interface elements to indicate and manipulate between the two modes (e.g., a radio button). Another option would be to keep the current, modeless interaction by altering playback to be always in the compressed domain. If normal playback is required in this scenario, the user must first decompress completely, and then play. If the user could also seek to any point, regardless of compression level, this might afford the best of both worlds.

We close by recounting a behaviour of one of the participants while performing task 1, browsing for a scene in a familiar movie. This participant invited one of the author's to sit with him, and verbalized his actions. He was hunting for a scene in Shrek. He knew only that the scene in question lay in the second half of the movie. He began by compressing the movie fully. He decreased the compression level gradually. The movie's climax and major events appeared in the compressed domain. He noted their appearance, and then continued to decompress the movie until the next tier of dramatic events began to appear. At this point he used the seek bar to observe one that had appeared at about the two-thirds mark. It was the scene he was looking for.

	Discrete	Continuous	Undecided
Preferred Colourization	**7**	1	
	Increased intensity	Increased height	Undecided
Preferred Compression Vis.	1	**4**	3
Interaction Element Votes			
Colour-coded interest		5	
Compression		6	
Compressed-domain playback		3	
Compression slider		6	

Table 2. Results for vis. of interest and compression effect.

5. CONCLUSION

We extend Temporal Semantic Compression for interactive video browsing, incorporating new interest metrics derived from user interaction statistics. Further, we visualise the interest measures and corresponding video summary, making it easier for the user to understand how the summary was produced. We perform a user study to demonstrate the power of the browsing paradigm and report design implications.

6. REFERENCES

[1] A. Carlier, V. Charvillat, W. Ooi, R. Grigoras, and G. Morin. Crowdsourced automatic zoom and scroll for video retargeting. In *Proc. ACM Int. Conf. on Multimedia*, MM'10, pp. 201–210, NY, NY, USA, 2010.

[2] K.-Y. Cheng, S.-J. Luo, B.-Y. Chen, and H.-H. Chu. Smartplayer: user-centric video fast-forwarding. In *Proc. Int. Conf. on Human factors in computing systems*, CHI'09, pp. 789–798, NY, NY, USA, 2009.

[3] A. Divakaran, K. Peker, and H. Sun. Constant pace skimming and temporal subsampling of video using motion activity. In *International Conference on Image Processing*, pp. 414–417, 2001.

[4] S. Drucker, A. Glatzer, S. D. Mar, and C. Wong. Smartskip: consumer level browsing and skipping of digital video content. In *Proc. SIGCHI Conf. on Human factors in computing systems*, CHI '02, pp. 219–226, NY, NY, USA, 2002.

[5] S. Greenhill, B. Adams, and S. Venkatesh. Interactively browsing movies in terms of action, foreshadowing and resolution. In *IEEE/ACM Joint Conf. on Digital Libraries (JCDL)*, June 2010.

[6] A. Haubold, P. Dutta, and J. Kender. Evaluation of video browser features and user interaction with VAST MM. In *Proc. ACM Int. Conf. on Multimedia*, MM'08, pp. 449–458, NY, NY, USA, 2008.

[7] A. Haubold and J. Kender. VAST MM: multimedia browser for presentation video. In *Proc. ACM Int. Conf. on Image and Video Retrieval*, CIVR'07, pp. 41–48, NY, NY, USA, 2007.

[8] W. Hürst. Interactive audio-visual video browsing. In *Proc. ACM Int. Conf. on Multimedia*, MM'06, pp. 675–678, NY, NY, USA, 2006.

[9] E. Ishak and S. Feiner. Content-aware scrolling. In *Proc. ACM Symp. on User interface software and technology*, UIST'06, pp. 155–158, NY, NY, USA, 2006.

[10] H. Lee and A. Smeaton. Designing the user interface for the físchlár digital video library. *Journal of Digital Information*, 2(4), 2006.

[11] K. Peker and A. Divakaran. Adaptive fast playback-based video skimming using a compressed-domain visual complexity measure. In *In IEEE Int. Conf. on Multimedia and Expo*, pp. 2055–2058, 2004.

[12] S. Pongnumkul, J. Wang, G. Ramos, and M. Cohen. Content-aware dynamic timeline for video browsing. In *Proc. ACM Symp. on User Interface Software and Technology*, pp. 139–142, NY, NY, USA, 2010.

[13] K. Schoeffmann, M. Taschwer, and L. Boeszoermenyi. The video explorer: a tool for navigation and searching within a single video based on fast content analysis. In *Proc. ACM SIGMM Conf. on Multimedia Systems*, pp. 247–258, NY, NY, USA, 2010.

[14] Y. Song and G. Marchionini. Effects of audio and visual surrogates for making sense of digital video. In *Proc. of the SIGCHI Conf. on Human factors in computing systems*, CHI'07, pp. 867–876, NY, NY, USA, 2007.

[15] B. Truong and S. Venkatesh. Video abstraction: A systematic review and classification. *ACM Trans. Multimedia Comput. Commun. Appl.*, 3, February 2007.

[16] W. Willett, J. Heer, and M. Agrawala. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 13:1129–1136, 2007.