Temporal semantic compression for video browsing

Brett Adams[*], Stewart Greenhill, Svetha Venkatesh
Department of Computing
Curtin University of Technology
GPO Box U1987, Perth, 6845, W. Australia
{b.adams,s.greenhill,s.venkatesh}@curtin.edu.au

ABSTRACT
Temporal Semantic Compression is a novel paradigm for browsing and playing video data, which satisfies dynamic compression levels via time-based measures of interest to guide where to resample a video. This enables interaction with a dynamic zoom factor, which is mapped to a scroll wheel or alternative 2D gesture. The result fuses automatic semantic analysis and human attention applied to the activity of browsing. We implement one version of a TSC browser, the Temporoscope–temporal analogue to a telescope–which uses movie tempo or its derivative to infer interest and drive compression, and a 2D interface to simultaneously control position and compression. The screen footprint contains a level of context that varies with compression level, and sits taxonomically between parameter-heavy expert workbenches and dumb VCR-like linear subsampling.

ACM Classification Keywords
H.3.1 Information Storage and Retrieval: Content Analysis and Indexing; H.5.1 Information Interfaces and Presentation: Multimedia Information Systems

General Terms
Algorithms, Human Factors, Experimentation

Author Keywords
Video browsing, Media aesthetics, Compression

INTRODUCTION
Video is the medium of our day. On-demand video warehouses, such as YouTube, are a defining aspect of Web 2.0, fuelling explosive growth in the consumption and creation of video. It will also figure prominently in an increasingly mobile context. Gartner predicts "close to half a billion [Mobile TV] subscribers worldwide" by 2010. In contrast, user interfaces for video remain a deeply embedded anachronism: The standard set of interactions–play, pause, rewind and fast forward–arose historically to support the non-random access technology of tape. While random access has been added to this set of interactions, via chapter markers or slider bars, it is of marginal help for the kind of information or entertainment foraging behaviours encouraged by the amount of video data accessible today.

Motivated by the intuition that when faced with the problem of displaying restricted amounts of video, an interest factor should guide the resampling process, we present a novel approach to browsing and playing video data, termed Temporal Semantic Compression (TSC). A TSC browser takes as input any time-based measure of *interest* or *information* of a video, and uses this to resample the original video to achieve a dynamically variable compression or zoom factor (we use the terms interchangeably) for interactive consumption. Metrics are ideally drawn from automatic media understanding algorithms, and may be generic, such as motion, or specific to a genre, such as *excitement* for sports [6], *anxiety* in smart home surveillance [13], news story change, *attention* for home video [12], dramatic structure [2], or their derivatives. The zoom factor is mapped to a gesture, such as a scroll wheel, so the browsing/playback experience is entirely interactive. We experiment with a 2D spatial control, where the horizontal axis controls position, and vertical the amount of semantic compression. The paradigm yields the following properties:

- Genericness – any measure of interest can drive compression. E.g., measures more appropriate to a video's genre may be plugged in during browsing.
- Fitness – seamless compression enables *modeless* interaction stretching from a single, static representation of an entire video, to normal playback, suitable for non-experts and browsing behaviour that "blur[s] the boundary between browsing and playback"[8].
- Scope – TSC browsers can have a screen footprint that is a drop-in replacement for existing video widgets. Server-side computation of interest measures leave only simple computation at the client, suitable for constrained settings like mobile media players.

TSC browsing is an example of computer-as-aide. The work of Liu and Kender [11] is closest to ours, with their framework for semantic video compression. It used a frame pair distance measure, together with a surrounding context of four frames, as criteria for leaking redundant frames from a fixed size buffer. We go much further by (i) allowing any time-based function, including the rich class of semantic functions that take an entire video for their domain, and even user-sensitive parameters for personalization, and (ii) marrying the compression action to existing browsing/playback interaction with a single additional input–temporal zoom–able to be interactively altered in concert with viewing. The zooming metaphor is a well known technique for allowing a user to investigate in detail while maintaining overall orientation and traversal of large amounts of data. As such,

[*]Project funded by the Australian Research Council, and Curtin University of Technology Fellowships Scheme.

a TSC browser sits taxonomically between an expert workbench (with many parameters to set and a large screen footprint), and a dumb terminal (linear subsampling via FF or REW). The significance of the approach lies in the sheer utility of video browsers: Hundreds of millions of videos are downloaded online alone each day.

BACKGROUND

The preeminence of the video medium motivates research in both automated media understanding as well as HCI, but successful fusion of the media-centered and user-centered perspectives has been elusive.

We first consider approaches arising from an attempt to understand the *content* of video data. There exists a host of techniques to abstract video, which may be taxonomized based on whether they yield a static or moving summary (e.g. storyboards or mosaics vs. video skims), their setting (e.g. couch [5] and desktop [9]), their degree of genre specificity (e.g. [10] feature film, [14] home video), and so on. A fundamental difficulty arises from attempting to compress in time a temporal medium. Most approaches require a high cognitive load and time to learn (appropriate to their settings) that does not translate to a generic browsing setting, which might entail lower concentration or reduced interaction complexity. Crucially, most approaches have at best coarse-level interactivity: Parameters are set and an abstract is produced batch. Without a human in the loop, the algorithms fail to cross the semantic gap, which has motivated many to target specific video genres and the effective heuristics they allow. Their primary goal is often to enable the user to assimilate maximal information in mimimal time.

On the user side of the equation, HCI research addresses how the video-related information is *received* and manipulated. This includes focii such as perceptual aspects like layout and spatial vs. temporal tradeoffs [4], interaction aspects like complexity and cognitive loading [7], and user situation and device constraints [3]. The interaction mode is usually assumed to be a 'consumer' setting, but this increasingly encompasses a range of devices (e.g. desktop computer, smartphone, media-player, DVRs, ITV) and consequently, interaction modes (e.g. mouse, stylus, touch, button). We need interaction paradigmd that readily map across these contexts, which the simple play and FF interface does. These approaches assume an interactive setting, but have less emphasis on computational understanding as an aid to media browsing. They often seek to lower viewer cognitive load or interaction complexity.

From this perspective we can see that the media-centered and user-centered approaches are complementary: the cognitive demands on a viewer or the complexity of interaction with a video device may be attenuated by offloading it to algorithms, and the weaknesses of those algorithms (particularly when aimed at extracting high-level semantic information) can be addressed by situating them in a UI that allows the user to assess content in view of their goals rapidly, and only when required: high opportunity, low deployment human-in-the-loop.

SYSTEM OVERVIEW

Infinitely many browsers might implement semantic compression which differ on design parameters, such as where interest measures are computed (e.g. server- or client-side), how the compression factor is input (e.g. separate or composite gesture), and how interaction and video context are visualized (e.g. static or variable sized overlays). We emphasize a design that is a drop-in replacement for existing video widgets on the desktop or screen-constrained settings using a single interaction gesture. We outline one instantiation of a TSC browser, the *Temporoscope*,[1] where the time-base can be continuously varied in a semantically meaningful way.

Browser

The Temporoscope looks like any default video browsing widget, with a playback area dominating screen footprint, and the addition of two elements:

- A "compression control" that allows the compression factor to be continuously varied between 0 and 1.
- A "context area" showing a variable amount of temporal context (the periphery of the spyglass). This consists of a set of thumbnails depicting key-frames preceding and succeeding the current shot, and is overlaid when the position/compression control is used.

Position in the video stream and compression factor are controlled by a single gesture: The playback area can be clicked in with the mouse or stylus, and the x-axis is mapped to position (frame number) and the y-axis is mapped to compression ratio (similar to the ZoomSlider [7]). Decoupling the compression factor and assigning it to a scroll wheel or jog button are an obvious conventional alternative. Compression factor is used to resample the original video dynamically, and the resulting context about the current position of the subsampled video is displayed to the user in thumbnails. At 100% compression, the displayed context is simply the most interesting key-frame. We now discuss an example interest measure and the sub-sampling process.

Semantic Analysis

A TSC browser requires a measure of interest or information, calculated as a function of time (frame or shot number) over an entire video. The measure ascribes relative importance to every frame, and is used as the criterion for resampling the video to achieve a given compression factor. In the parlance of lossy compression, regions of higher importance correspond to salient content, and should be preserved after compression.

'Importance' is a subjective quality, relative to the viewer's specific intention in viewing a video. Useful measures for a range of genres are cited above, which are based on structures *inherent* to videos of their genre. As such they naturally support large classes of browsing goals. How to choose an appropriate measure at browse time is

[1]Tongue firmly in cheek: A Temporoscope is the temporal analogue to a telescope, wherein a user may scan large swathes of landscape, punctuated by zooming on objects of interest to obtain more detail at will.

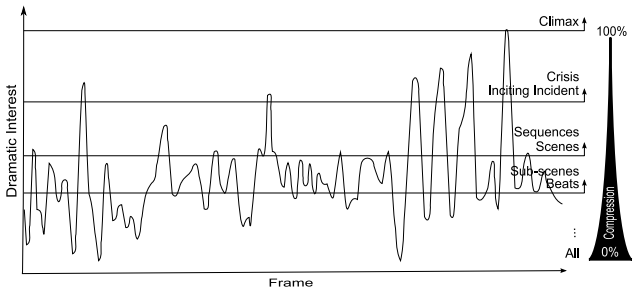

Figure 1. Compression on interest measure: as the compression threshold is raised, video is seamlessly filtered of less significant dramatic content.

beyond the scope of this paper, except to say that measures appropriate to large classes of genre will make good defaults, automatic genre classification will help choose more appropriate measures for a given video, and measures that explicitly include viewer settable parameters will open the way for highly personalized browsing.

We demonstrate TSC browsing with a measure proven useful across many genres: *tempo* [1], and its aural counterpart, which were formulated based on the notion that filmmakers manipulate an audience's sense of time by controlling the amount of *information* thrust at them: action, music, and dialog can each be increased or withheld with the effect of making a movie race or crawl. It is thus a natural signal to compress. Tempo is tied to relatively low-level physiological effects, and is a useful measure for many genres, including those with very little mediation, such as home video or surveillance. Raw tempo provides an interest measure that privileges action. Additionally, we take the absolute value of the derivative of combined visual and aural tempo as an indicator of dramatic interest (i.e., underlying narrative events that give rise to the ebb and flow of action). Figure 1 is a mock-up of dramatic interest for *The Truman Show*. A number of thresholds depict the amount of compression and corresponding dramatic structures preserved. At 100% compression the most salient dramatic incident, the climax, is preserved.

Tempo is calculated as per the equation of [1], and is formulated as follows. Let F be a set of frame-level features. In this study, $F = \{pan, tilt, volume\}$ where *pan* and *tilt* are camera motion parameters, and *volume* is the mean audio volume. For each feature, we define the global mean μ_i and standard deviation σ_i. The tempo T is defined as:

$$T(n) = \alpha(W(s(n))) + \sum_{i \in F} \frac{\beta_i(\mu_i(n) - \mu_i)}{\sigma_i}$$

Where $s(n)$ is length of shot n, and $\mu_i(n)$ is mean value of feature i over shot n. W is the shot normalisation function described in [1], which tapers linearly from 1 at 0 to 0 at the median shot length, then asymptotically approaches -1 at a shot length below which 95% of the shots lengths are distributed. α and β_i are weights for individual features. Nominally, $\alpha = 1$, and $\beta_i = \frac{1}{3}$. $T(n)$ is smoothed using a Gaussian filter ($\sigma = 2$) and the derivative $T'(n)$ is computed using a recursive filter.

Figure 2. Browser playback area (top) and context area (bottom). Overlay is only displayed when position or compression level is changed.

Semantic Compression

We define *semantic compression* as the process of using high-level semantic information to control the amount of information presented to the viewer. The user defines a *compression factor* $0 < f <= 1$, and the system selectively discards video frames so that if there were intially N frames, fN frames remain after compression. A *compression function* manipulates the video time-base, mapping each frame in the compressed video to one in the original video. They have a range $[0, fN - 1]$, and domain $[0, N-1]$, and normally increase monotonically. For simple linear compression, we have:

$$linear_f(n) = fn$$

corresponding to "$1/f$ times" fast-forward playback.

There are two fundamental ways in which to compression can be done: temporality-preserving, where frames or shots are dropped, but the video is always played at a normal rate, or atemporal in which, in addition to shots/frames being dropped, playback rate can be altered. Although we experimented with the latter approach, this paper focuses on the first approach. Given a compression factor, a duration for the compressed video is calculated, τ. All N shots (or uniformly distributed contiguous chunks, in the case of video genres that do not employ montage) are ranked according to interest, $s_1 \ldots s_N$. The compressed video is then simply the set of shots satisfying $\sum_{i=1}^{k} dur(s_i) \approx \tau$, ordered according to their original relative positions. A variant of this algorithm is to choose fractions (potentially lower bounded to a floor of, say $l = 1$ second) of shots, i.e., $\sum_{i=1}^{k} (f \times dur(s_i))\rfloor_l \approx \tau$. This algorithm supplies a larger number of shot fragments, at the expense of shot context. The playback rate of the chosen frames by ei-

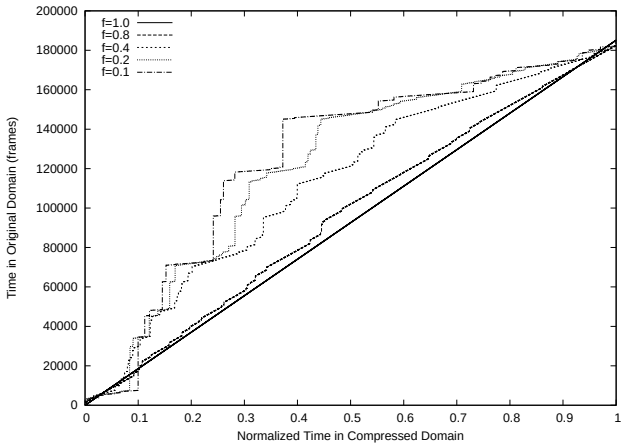

Figure 3. Sub-sampling with varying compression.

ther method is normal, thus preserving the medium's temporality, and the result is a version of the video filtered of all but the top $f\%$ of significant highlights.

Results

We experimented with the Temporoscope using both action and drama interest, on a number of movies and a 4 hour block of daytime TV (including a range of genres, such as news, commercials, cartoons, and talk shows). Feature extraction was cached prior to browsing. A demo can be found at impca.cs.curtin.edu.au/cma.php.

Figure 2 shows the position + compression control in use. Varying the compression amount causes the compression function to be recalculated, which potentially adds or removes shots from the context display. When the function changes, the playback position is adjusted to the closest frame in the new function. The user can dynamically adjust the amount of context displayed by resizing the context area. The video area displays the frame at the current time in the compressed domain, but the context area shows a window into past and future centered around this time. Informal trials indicated the compression concept is comprehensible and usable. Mapping compression to the familiar scroll wheel may reduce the slight learning curve even further.

Figure 3 (top) displays the mapping from the input time (vertical axis) to the compressed domain (horizontal axis) for different levels of compression for *The Matrix*. Figure 4 shows the context area overlay for different levels of compression: the top row shows the most compressed version (shows entire escape scene bracketed by the preceding and following scenes) whilst each row below represents lesser compression, and thus more detail. We note here that varying σ causes more or less shot fragmentation. E.g., higher σ gives a smoother interest measure, and thus results in selection of larger contiguous clusters of shots. The result is *more* shots in the context area gathered about *less* movie events.

CONCLUSION

This work presents a browsing paradigm that takes an arbitrary interest factor and uses it to guide shot selection to achieve dynamically changing levels of compression. The single gesture interface allows the user to seamlessly navigate video, dynamically varying position, compression and context. We have implmented and demonstrated the usefulness of one such implementation using tempo to drive the semantic compression.

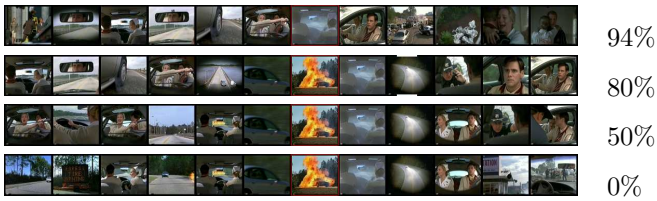

Figure 4. Context overlay changes with compression, centered on *Escape Sequence*: 94% Sequence bracketed by two significant events; 80% Includes two sub-sequence events, fire and escape on foot; 50% Detail includes dialog giving rise to escape on foot; 0% Original video.

REFERENCES

1. B. Adams, C. Dorai, and S. Venkatesh. Towards automatic extraction of expressive elements from motion pictures: Tempo. *IEEE Transactions on Multimedia*, 4(4):472–481, Dec. 2002.
2. B. Adams, C. Dorai, S. Venkatesh, and J. Bui. A probabilistic framework for extracting narrative structure and semantics in motion pictures. *Multimedia Tools and Applications*, 27(2):195–213, November 2005.
3. M. Barbieri, G. Mekenkamp, M. Ceccarelli, and J. Nesvadba. The color browser: a content driven linear video browsing tool. In *IEEE International Conference on Multimedia and Expo*, p.627–630, Aug 2001.
4. A. Divakaran, C. Forlines, T. Lanning, S. Shipman, and K. Wittenburg. Augmenting fast-forward and rewind for personal digital video recorders. In *Consumer Electronics, ICCE*, pp.43–44, Jan. 2005.
5. 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*, p.219–226, 2002.
6. A. Hanjalic. Adaptive extraction of highlights from a sport video based on excitement modeling. *IEEE Transactions on Multimedia*, 7(6):1114–1122, 2005.
7. W. Hurst and P. Jarvers. Interactive, dynamic video browsing with the zoomslider interface. In *IEEE International Conference on Multimedia and Expo*, 2005.
8. H. Lee and A. Smeaton. Designing the user interface for the fischlar digital video library. *Journal of Digital Information*, 2(4), May 2002.
9. F. Li, A. Gupta, E. Sanocki, L.-W. He, and Y. Rui. Browsing digital video. In *CHI '00: SIGCHI conference on Human factors in computing systems*, p.169–176, New York, NY, USA, 2000. ACM Press.
10. Y. Li, S.-H. Lee, C.-H. Yeh, and C. Kuo. Techniques for movie content analysis and skimming: tutorial and overview on video abstraction techniques. *Signal Processing Magazine, IEEE*, 23(2):79–89, 2006.
11. T. Liu and J. Kender. Time-constrained dynamic semantic compression for video indexing and interactive searching. In *Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition*, p.II–531–II–538, 2001.
12. Y.-. F. Ma, L. Lu, H.-. J. Zhang, and M. Li. A user attention model for video summarization. In *Proc. of the 10th ACM Int. Conf. on Multimedia*, p.533–542, Nov. 2002.
13. S. Moncrieff, S. Venkatesh, G. West, and S. Greenhill. Multi-modal emotive computing in a smart house environment. *Pervasive and Mobile Computing*, In Press, Corrected Proof, 2007. To Appear.
14. Z. Pan and C.-W. Ngo. Structuring home video by snippet detection and pattern parsing. In *MIR '04: Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, pages 69–76, New York, NY, USA, 2004. ACM Press.