

Hyper-Hitchcock: Towards the Easy Authoring of Interactive Video

Frank Shipman, Andreas Girgensohn, Lynn Wilcox

FX Palo Alto Laboratory, 3400 Hillview Avenue, Palo Alto, CA 94304, USA

shipman@cs.tamu.edu, andreasg@fxpal.com, wilcox@fxpal.com

Abstract: To simplify the process of editing interactive video, we developed the concept of “detail-on-demand” video as a subset of general hypervideo where a single button press reveals additional information about the current video sequence. Detail-on-demand video keeps the authoring and viewing interfaces relatively simple while supporting a wide range of interactive video applications. Our editor, Hyper-Hitchcock, builds on prior work on automatic analysis to find the best quality video clips. It introduces video composites as an abstraction for grouping and manipulating sets of video clips. Navigational links can be created between any two video clips or composites. Such links offer a variety of return behaviors for when the linked video is completed that can be tailored to different materials. Initial impressions from a pilot study indicate that Hyper-Hitchcock is easy to learn although the behavior of links is not immediately intuitive for all users.

Keywords: Hypervideo, interactive video, video editing, video analysis, video exploration.

1 Introduction

There is a need for editors that enable a wider community of interactive video authors. Entertainment companies look to interactive video as a natural result of merging television, interactive fiction, and computer games. In addition, the growing use of digital cameras (video and “still”) to capture short video snippets makes home authoring of interactive video an application that is likely to emerge.

There is a trade-off between expressiveness and ease of use in choosing among current tools to support authoring of interactive video. With iDVD (Apple Computer), authors create opportunities for viewers to select among pieces of video via the menu-based interaction found on DVDs. To allow viewers to interact directly with video, authors generally resort to a more general purpose multimedia authoring environment, such as Director (Macromedia). These examples mark extremes for the interactive video expressiveness and ease of use spectrum.

One form of interactive video, hypervideo, restricts the interactions to following links between videos. It allows a greater range of interactions than chapter selection for DVDs but is more restrictive than general multimedia and thus simpler to author. To make authoring even easier, we chose to investigate a simpler form of hypervideo, *detail-on-demand* video, which only supports one link at any given time rather than allowing for multiple link anchors in different areas of a video frame. Viewing detail-on-

demand video combines interaction characteristics from browsing the Web and changing channels on TV. As the viewer watches a video, the player indicates when links are available and presents labels for them (see Figure 1). After pressing a button, the link destination video plays until completion, at which time the original video continues.

We have designed Hyper-Hitchcock, an authoring and playing interface that supports authors in creating detail-on-demand video by providing a direct



Figure 1: Player with Navigation Buttons and Link Indicators in Timeline

manipulation environment in which authors can combine video clips and place hyperlinks between them. Unlike traditional hypervideo links that jump to a different part of the video, links in detail-on-demand video also express what should happen when the linked video clip finishes playing or when the user presses the “back” button. These link behaviors can be used to establish relationships between the link source and destination. For example, if the link destination is a longer version of the source, there is no need to watch the rest of the shorter version.

The next section describes the representation of video and links in detail-on-demand video. HyperHitchcock, the authoring and viewing environment for detail-on-demand video, is then described. Following this we present some initial user experiences and conclusions.

2 Detail-on-demand Video

Hypervideo allows viewers to navigate between video chunks. Applications of hypervideo include educational environments (Guimares et al. 2000) and narrative storytelling (Sawhney et al. 1997). General hypervideo allows multiple simultaneous link anchors on the screen, e.g., links from characters on the screen to their biographies. This generally requires anchor tracking — tracking the movement of objects in the video that act as hotspots (source anchors for navigational links). As a result, research on supporting hypervideo authors has emphasized methods for defining and tracking source anchors (Hirata et al. 1996; Smith et al. 2000). Research on supporting hypervideo viewers includes algorithms for anticipating navigation among multiple available links (Grigoras et al. 2002).

We have chosen to investigate *detail-on-demand* video as a simpler form of hypervideo, where at most one link is available at any given time. Authoring such video can be supported in a direct manipulation video editor rather than requiring scripting languages or other tools that are unsuitable for a broad user base. At its simplest, the author selects a segment of the edited video for which a link will be active and the video sequence that will be shown if the viewer takes that link. By removing the need to define and track hot spots in video, the authoring and viewing interfaces can be simplified.

This representation creates a natural mechanism for authoring “how to” videos where viewers can get the level of explanation they need. With the main video stream presenting the topic/process at a more abstract or coarser-grained level, the viewer can navigate to view the aspects of the topic for which they need more help. This improvement over current linear access how-to videos will save the viewer’s time and make it possible for video guides to be appropriate to a wider audience. A second application of

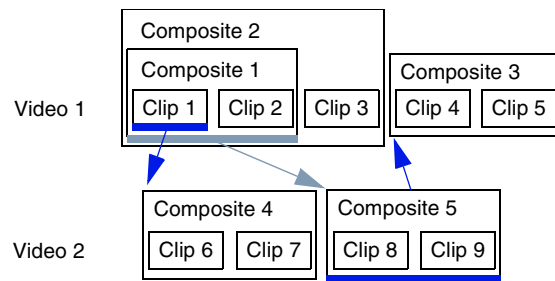


Figure 2: Hierarchically-Organized Videos and Links

detail-on-demand video is the family video Christmas card or DVD, where the main video stream is an overview of the family’s activities for the year. In this use, viewers can navigate to get more footage about the activities of interest, perhaps the family reunion, the vacation, or the birthday party.

Detail-on-demand video is accessible to less technically inclined authors through its representation of interactive video that emphasizes ease of learning and use over the expressiveness of the resulting video’s interactive behavior. The representation’s primary features are navigational links between hierarchical video composites and link properties defining link labels and return behaviors.

2.1 Hierarchical Video with Links

Composing detail-on-demand video consists of authoring one or more linear video sequences and creating links between the elements of these sequences. Each video sequence is represented as a hierarchy of video elements. Segments of source video (clips) are grouped together into video composites, which themselves may be part of higher-level video composites. Links may exist between any two elements within these video sequences. The source element defines the source anchor for the link — the period of video playback during which the link is available to the viewer. The destination element defines the video sequence that will be played if the viewer takes the link. The source and destination elements specify both a start and an end time.

To keep the interface simple, detail-on-demand video allows only one link to be attached to each element. Figure 2 shows three links between two video sequences. The links are from Clip 1 to Composite 4, from Composite 1 to Composite 5, and from Composite 5 to Composite 3. Also, at most one link is made available to the viewer at any time. If multiple levels of the hierarchy have links that overlap, the viewer is provided with the most specific link — that is the link attached to the innermost element in the hierarchy. In the example in Figure 2, viewers see the link out of Clip 1 to Composite 4 at the beginning of Video 1 during Clip 1 and the link from Composite 1 to Composite 5 during Clip 2.

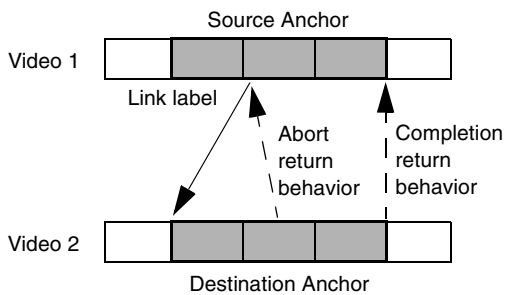


Figure 3: A detail-on-demand video link includes source and destination anchors, link label, and return behaviors.

2.2 Link Labels and Behaviors

Links in a hypervideo can have a number of characteristics that change their impact on the resulting presentation. Two such characteristics are the label shown to the viewer as an indicator of the content of the link and the behavior when returning from links (see Figure 3). There are two independent link behaviors: (1) what happens when the destination sequence of a video link finishes playing, and (2) what happens when the viewer of the destination sequence ends its presentation before it is finished. There are a number of possibilities for what the video player should do in both cases.

The detail-on-demand video author can choose from a predefined set of behaviors. The four options are (1) play from where the viewer left the video, (2) play from the end of source anchor sequence, (3) play from beginning of source anchor sequence, and (4) stop playback. Picking up from where the viewer left the video sequence (or perhaps shortly before that point to provide some context), is a common option as it treats the content at the other end of the link as independent of the currently playing content. Playing the video from the beginning of the source anchor may be appropriate for training tasks where the more detailed knowledge is a prerequisite for understanding the higher-level process. By playing the video sequence from the end of the source anchor, the side trip becomes an alternate route, enabling the video to provide both long and short descriptions of the same concept. This is the “completion return behavior” for the link in Figure 3. Finally, a destination anchor that presents concluding material might stop video playback altogether.

The second link characteristic is determining what to do when the viewer decides to return early from a link destination. In hypertext, like the Web, pressing the “back” button takes the user back to where they last took a navigational link. Similarly, the “return” button in detail-on-demand video would most often return to the location of link navigation, as in Figure 3. In addition to this “intuitive” behavior, the same options for where to restart playback of the source anchor apply as above.

2.3 Hierarchical Access to Video

A variety of interfaces for accessing video make use of an explicit or inferred hierarchy for selecting a starting point from which to play the video. These vary from the standard scene selection on DVDs to selection from hierarchically structured keyframes or text outlines in a separate window (Rui et al. 1998; Myers et al. 2001). Selecting a label or keyframe in a tree view is used to select a point for playback.

The primary difference between interfaces supporting hierarchical access to video and detail-on-demand video is the detail-on-demand viewer may request additional detail while watching the video rather than having to use a separate interface such as keyframes or a tree view. Also, the hierarchical representation of a video does not include semantics beyond simple hierarchical composition. The links in hypervideo have labels and a variety of behaviors for when the link’s destination anchor finishes playback or when the user interrupts playback. Links between clips or composites in a hypervideo support the viewing of additional detail and the automatic return to the main video thread.

3 Hyper-Hitchcock

Hyper-Hitchcock is a direct manipulation environment for authoring and viewing detail-on-demand video (see Figure 4). The authoring tool is characterized by three panes: a selection pane in the upper left, an authoring workspace on the bottom, and a tree view of the workspace contents in the upper right.

The selection pane has the same function as in the original Hitchcock video editor (Girgensohn et al. 2001) and displays a hierarchically arranged set of keyframes representing clips from the source video. Video takes, defined by camera on-off pairs, are automatically divided into clips based on camera motion and changes in brightness. Additionally, the system uses the analysis of movement and brightness to identify default in/out points for the clip such that the highest quality clip segment is included. Metadata about the clip, including the take, the length of the overall clip, the default segment, and the computed “goodness” of the video are indicated on information bars that appear at the bottom of clips. Video clips are arranged in piles grouped by different criteria such as recording date. Users can flip through a pile and locate clips by moving the mouse over the pile. Video clips have associated tool tips that reveal information such as the clip and take identifier or the recording date. The timeline above the selection pane displays the coverage of the pile in the source video and the start of the clip under the mouse.

Authors can drag clips from the selection area into the workspace. There, the length of the segment of clip to be included in the edited video may be changed by resizing the image of the clip. As with

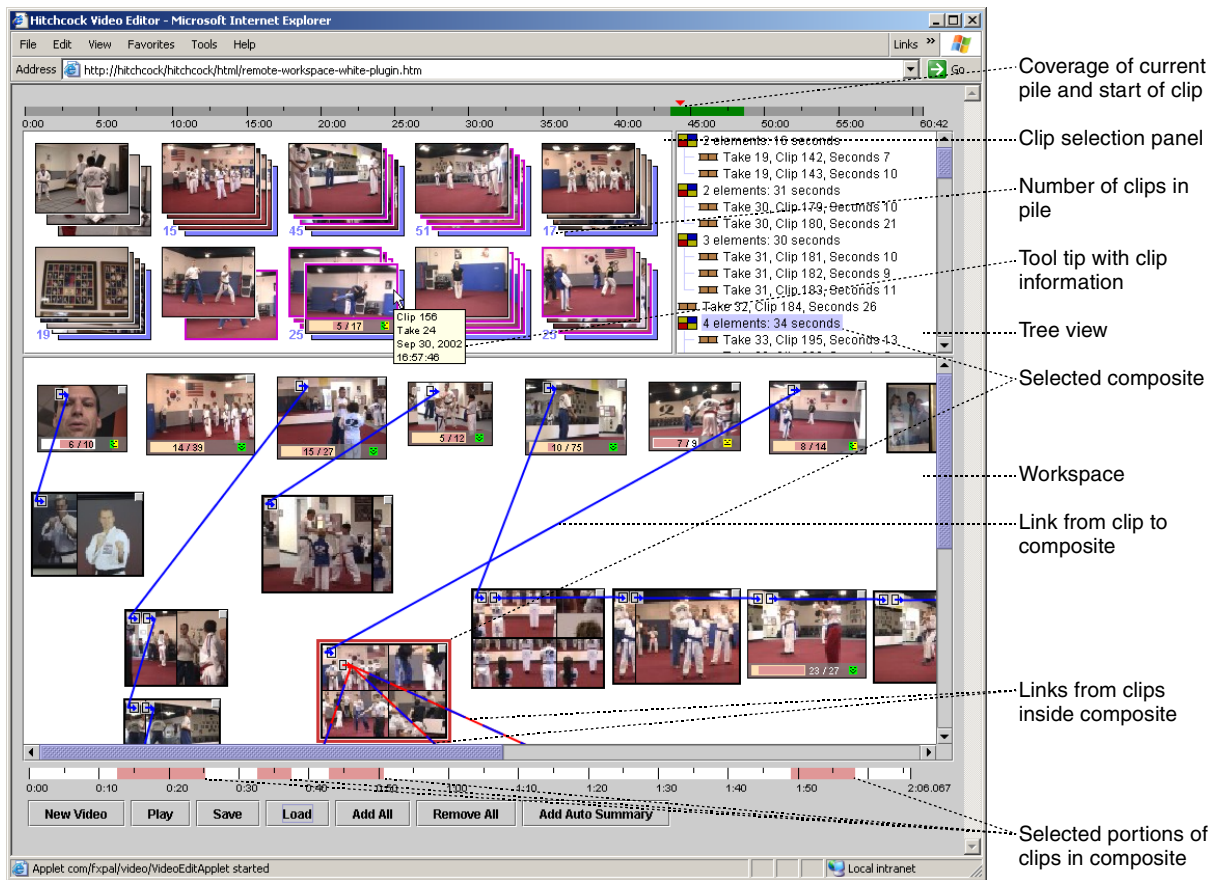


Figure 4: Hyper-Hitchcock

Hitchcock, the in/out points are automatically adjusted to select the highest quality segment of the desired length by default. The timeline below the workspace displays the in/out points of the selected video clips. If only one clip is selected, the author can lock and move the in and/or out points in the timeline. When several clips or composites are selected, the timeline displays the included portions of all selected clips (as shown in Figure 4).

3.1 Video Composites

Clips may be grouped into video composites, which in turn may be part of other (higher level) composites. This hierarchy of clips is presented in the tree view (see Figure 5). The workspace and the tree view are alternate representations for the same collection of video clips and composites. The tree view reveals the deep structure of a composite whereas the workspace displays keyframes for up to four clips included in a composite. The workspace also visualizes links going into and out of a composite.

Selecting clips and composites in either view also selects them in the other view. Because there is no explicit representation in the workspace for clips included in a composite, the enclosing composite is highlighted in a fainter color when a nested clip is

selected in the tree view. This is especially important during playback when the video player selects clips as they are played.

When the author creates a composite, the system generates a visualization for the composite made up of parts of two to four of the keyframes from the component clips (see Figure 7). The first and last keyframes are always included as these are most valuable for the author when placing the composite in a sequence with other workspace elements. The

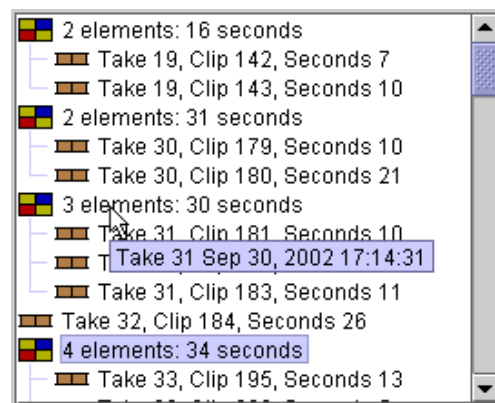


Figure 5: Tree View Showing Composites

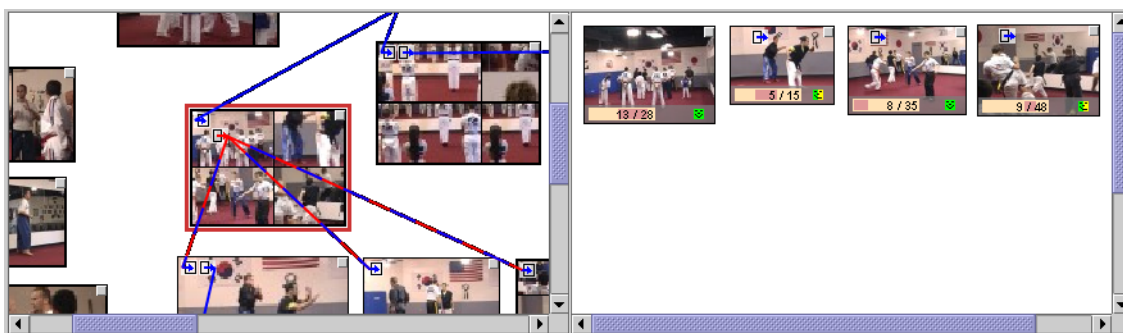


Figure 6: Split Workspace for Editing a Composite

two longest of the remaining clips are also represented. The areas of the included keyframes are adjusted relative to the lengths of the clips by moving the horizontal and vertical borders between them.

As with clip keyframes, the size of the composite visualization indicates its length. Adjusting the size of the visualization will cause the in/out points of the component clips to be altered (while still respecting in/out points locked by the author) so as to keep the relative lengths of the component clips the same until this is no longer possible. As with individual clips, the system automatically selects the highest-quality segment for each individual clip unless the user locked in/out points. When resizing a composite, the timeline below the workspace shows which portions of the included clips are being used.

Once a composite is created, the author may wish to add, remove, resize, or reorder components. When an author opens a composite for editing, the workspace pane splits into two halves, with the overall view in the left half and the composite in the right (see Figure 6). The author edits the composites in the normal manner and may drag clips between the two halves. While a composite is being edited, it is also possible to modify links originating or terminating at clips included in the composite.

Within the composite view, and when playing a selection of elements in the workspace, an implicit order must be determined for the elements. By com-

paring the width and the height of the extent of the area including all the elements, the system decides whether to use horizontal or vertical position to order the elements. This works well for horizontal and vertical lists of elements, for example the top row of elements in the workspace in Figure 3 or the composite elements on the right side of Figure 6. In the future, implicit orderings could be based on a spatial parsing algorithm that can identify more complex spatial structures, such as lists of lists (Shipman et al. 1995). The same implicit order is used when playing several selected clips or composites from the workspace. This allows for the construction of the main video sequence as a horizontal or vertical list with links branching off from it.

3.2 Navigational Links

Navigational links can be created between any two elements in the workspace. Once created, links are visualized as colored arrows between clips or composites. Line placement provides information about whether the link is into or out of an element in the workspace and the location and color of the link indicate if the link is connected to the whole element in the workspace or one of its components. Figure 7 shows a close up of the arrows in and out of one of the composites in Figure 4. In this case there is one incoming link to the whole composite, represented by the blue arrow at the top left, and three red/blue links from components of this composite to other elements in the workspace. Links can be authored to form cycles, counterpoints, tangles, mirror worlds, and all the other common patterns of hypertext (Bernstein 1998).

Figure 8 shows the dialog where authors can change properties of links. Labels may be attached to links to provide viewers with information about the link destination's content. Metadata about the destination is used to generate default labels for links. These defaults provide take, clip, and length information for clips and the number of elements and length for composites. Links have return behaviors that control where the video continues after the playback of a link destination completes or after the viewer presses the "back" button to return to the link source without

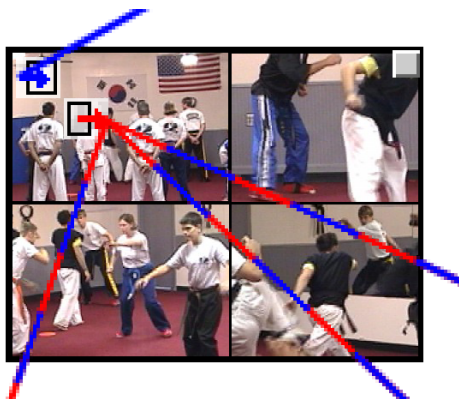


Figure 7: Links In and Out of Composite

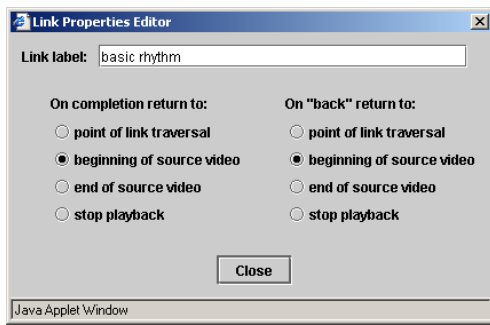


Figure 8: Link Properties

viewing the entire destination. Returning to the point where the side trip started is the default behavior but other options include returning to the start or the end of the link’s source anchor. It is also possible for the link’s return to stop video playback altogether.

In addition to the links to other video elements, each element in the workspace may have a link to external information. This is a URL attached to the element that can be used to provide the viewer with related materials in other media (e.g., text, still images). Determination of which link to external information is active follows the same process as for determining which video link is active.

3.3 Player

Viewing video in Hyper-Hitchcock combines interaction characteristics from browsing the Web and changing channels on TV. As the viewer watches a video, the player indicates when links are available and presents labels for them (see Figure 1). The viewer can follow the link to see the destination video or let the original video keep playing. The destination video will play until completion, at which time the original video will continue as indicated by the link’s return behavior. If the destination video is not of interest, the viewer can press a “back” button to return to the source video similar to backtracking in a Web browser.

Player Controls

Figure 1 includes close-ups of the interaction points for viewers of detail-on-demand video. Below the video playback region is the label for the link followed to get to the currently playing video. In Figure 1, that link is labeled “Capoeira”. Beneath that label are control buttons, a label for the currently available link (“basic rhythm” in Figure 1), a slider indicating the current point in the playback and the boundaries of links in the currently playing video, and the time into the currently playing video. The control buttons (from left to right) are for starting and stopping the video, taking links, aborting the playback of links, and retrieving external information. After taking the link in Figure 1, the main label would be replaced with “basic rhythm” and a new label would be dis-

played if a new link became available. The button for retrieving external information is grayed out in Figure 1 indicating that no external link is currently available to the viewer. Were one available and selected by the viewer, the video would keep playing while a separate browser window was launched and loaded with the indicated external information.

The video slider shows the length of the video sequence currently being played and the current position in that playback. It also shows when links will be active during that playback as blue shaded regions in the slider. The boundaries between links are indicated by darker blue lines in the slider. In the video playing in Figure 1, there is a period without any link and then three periods with links.

Transitions Between Video Sequences

When a viewer takes a link or returns from a link, a brief video icon is played to provide an indication to the viewer as to what has happened. Video icons are short (less than 1 second) video clips with distinctive audio tracks played in between the two pieces of authored video. These video icons were added because jumping straight from the source video to the destination video happened too quickly and confused viewers. There are different video icons for taking links, returning due to completion of the destination playback, and returning due to the user aborting playback.

Frequently, it would be preferable to use a transition including material from the source and destination clip when following a link. Examples for such transitions are fades to black, cross fades, and wipes. However, there are several problems that prevent us from using transitions in our player. First, transitions require render time. Our approach is to let the author play any part of the video being edited including linked portions without requiring any delay for rendering. We accomplish that by not modifying the source videos and instead controlling a video player plug-in and requesting it to skip to a different position in a video or a different video whenever the end of a video clip is reached or the viewer follows a link. Even if we were willing to accept a rendering delay, viewers could follow a link at any time during the playback of a clip so that one could not predetermine which part of the video in a link anchor to use in a transition. Transitions in this situation are only possible if they can be generated in real time by the player.

Visualizing a Web of Links

A potential difficulty for viewers is getting lost in the tangle of links in a complex hypervideo. To aid understanding of their context, the viewer may also keep open the workspace view of the hypervideo they are watching. As the player moves from element to element due to groupings of elements or links between elements, the currently playing element will be selected in the editor. This visual indication of the

viewers current location in the hypervideo reduces the disorientation possible when returning from multiple links within a small span of time.

4 Initial Impressions from Users

In preparation for a future user study, we conducted a pilot study with two participants to determine how users would interact with Hyper-Hitchcock. We created a tutorial that walked users through the different parts of the system. This tutorial was completed in about five minutes.

4.1 Types of Videos Created

After completing the tutorial, the participants of the pilot study were asked to create their own hypervideos from their own source material. One participant decided to tell a story about a Japan visit where hyperlinks went into parts of the story that would be of interest to viewers but not necessary for understanding the main story line. The other participant structured video from a mountain bike race such that different viewers could get more detail on the parts they would be most interested in. One such sidetrip included jumps the racers performed and another had more footage of his daughter (see Figure 9). The four video clips at the top of the figure represent the main story line. The three sidetrips show additional material about the drive, jumps of the racers, and footage of the participant's daughter. The last sidetrip allows the viewer to see more and more footage if desired.

4.2 User Feedback

We observed the participants of the pilot study, discussed system features with them, and asked them to fill out a questionnaire after they completed their video. Neither participant had problems using the system but one was initially confused about the fact that links returned to the point at which the viewer followed the link after completion of the playback of the linked video. He tried to achieve the same effect with an additional link and was happy that the system already took care of that.

The early questions were aimed at determining the amount of material included in the edited video (around one minute each; between 7.4% and 8.9% of the source video), the number of composites created (2 and 4), and the number of links created (5 each).

Both participants used link attributes, one to label the links and the other to define return points into the "main" story. Neither participant had use for the tree view. However, the first participant discovered that totalling the clip lengths in the tree view was one way to answer the question about the length of the selected video material.

Both participants liked the visualization of composites. One of them also praised the visualization in the timeline but would have liked to have a preview



Figure 9: Hypervideo Created by Study Participant

when making changes in the timeline. Both were satisfied with the representation of links in the workspace but one would have preferred larger link arrows and the other would have liked links to be highlighted when changing link properties.

When asked what they would use this form of video for, one replied that he would tell stories that would profit from extended side information. The other saw making his wife happy as the main benefit. He could edit the video without removing parts that she might like without having to keep them in the main video thread.

Regarding the most difficult aspect of the authoring experience, one of participants found the tracking of the logical paths of links difficult. The other found it difficult to decide what to keep and commented that audio would be an important deciding factor.

When asked for additional recommendation, one participant would prefer using the keyboard instead of the mouse for following links in the player. The other participant had several suggestions. First, he would like to see a label for where the "back" button would take him in the player. Second, he would like to have a confirmation dialog before ungrouping a composite with attached links to avoid losing the links. Finally, he would like to attach people's names to links to automatically play linear versions of the hypervideo customized to the viewer.

With this initial feedback for our system, we plan to address some of the problems and comments and then to conduct a formal user study.

4.3 Discussion

The experiences with these first two users provide support for the claim that Hyper-Hitchcock eases the authoring of detail-on-demand interactive video. The users did not appear to miss the additional expressiveness of tools like Macromedia Director. Instead, they had positive comments about our approach to hypervideo and came up with ideas for how to use such a system.

Issues identified by the pilot study include the lack of value of the tree view and some confusion

about the concept of side trips instead of links that jump to a new position in the video but do not return. The tree view, as a second perspective of the main workspace, may be redundant even though it provides a deeper view of the hierarchical structure. Further evaluation is necessary to determine if this is true as the two users only created composites with few clips and no nested composites during their short authoring sessions.

With respect to the return behavior of links, our users had preconceived notions about links that just take them to a different point in the video. One of the users attempted to explicitly create a return link and was pleased after being told that the system already took care of that. Because we introduce a new concept, we need to find appropriate visualizations for making that concept clear for hypervideo authors.

5 Conclusions

In this paper, we introduced the notion of detail-on-demand video as a flavor of hypervideo that simplifies authoring. In detail-on-demand video, a viewer can press a button to get more information about the current video sequence. This interaction style is well-suited for “how to” videos where the main video stream presenting the topic at a more abstract level and the viewer can navigate to view the aspects for which they need more help in greater detail.

Detail-on-demand video enables a wider range of authors through its representation of interactive video that emphasizes ease of learning and use over the expressiveness of the resulting video’s interactive behavior. We have designed Hyper-Hitchcock, an authoring and playing interface that supports authors in creating detail-on-demand video by providing a direct manipulation environment in which authors can combine video clips and place hyperlinks between them. To simplify editing, we introduced video composites as an abstraction for grouping and manipulating sets of video clips. Unlike traditional hypervideo links that jump to a different part of the video, links in detail-on-demand video also express what should happen when the linked video clip finishes playing. A variety of link return behaviors can be selected to suit different materials.

Navigational links in video present a new experience for most people and there are no consistent intuitions as to the behavior of these links. As such, Hyper-Hitchcock needs to be as clear as possible about the effects of links. Early hypervideo viewers will likely experience similar problems as with early hypertext users becoming “lost in hyperspace” or reaching dead ends. A final lesson from this work is that there is a rich design space for interactive video where the combined design of representation and interface can create useful and usable authoring and viewing environments.

Acknowledgements

We thank our study participants for taking time to create video footage and edit it with Hyper-Hitchcock.

References

- Apple Computer. iDVD. <http://www.apple.com/ivdv/>
- Bernstein, M. (1998). Patterns of Hypertext, in *Proceedings of ACM Hypertext '98*, ACM Press, pp. 21-29.
- Girgensohn, A., Bly, S., Shipman, F., Boreczky, J., and Wilcox, L. (2001). Home Video Editing Made Easy — Balancing Automation and User Control, in *Human-Computer Interaction INTERACT '01*, IOS Press, pp. 464-471.
- Grigoras, R., Charvillat, V. and Douze, M. (2002). Optimizing Hypervideo Navigation Using a Markov Decision Process Approach, in *Proceedings of ACM Multimedia*, ACM Press, pp. 39-48.
- Guimares, N., Chambel, T., and Bidarra, J. (2000). From Cognitive Maps to Hypervideo: Supporting Flexible and Rich Learner-Centred Environments, *Interactive Multimedia Journal of Computer-Enhanced Learning*, 2 (2), <http://imej.wfu.edu/articles/2000/2/03/>.
- Hirata, K., Hara, Y., Takano, H., and Kawasaki, S. (1996). Content-oriented Integration in Hypermedia Systems, in *Proceedings of ACM Hypertext '96*, ACM Press, pp. 11-21.
- Macromedia. Director MX. <http://www.macromedia.com/software/director/>
- Myers, B., Casares, J., Stevens, S., Dabbish, L., Yocum, D. and Corbett, A. (2001). A Multi-View Intelligent Editor for Digital Video Libraries, in *Proceedings of the ACM / IEEE Joint Conference on Digital Libraries*, ACM Press, pp. 106-115.
- Rui, Y., Huang, T.S., and Mehrotra, S. (1998). Exploring Video Structure Beyond the Shots. *International Conference on Multimedia Computing and Systems*, pp. 237-240.
- Sawhney, N., Balcom, D., and Smith, I. (1997). Authoring and Navigating Video in Space and Time, in *IEEE Multimedia Journal*, 4 (4), pp. 30-39.
- Shipman, F., Marshall, C., and Moran, T. (1995). Finding and Using Implicit Structure in Human-Organized Spatial Layouts of Information, in *Proceedings of Human Factors in Computing Systems (CHI'95)*, pp. 346-353.
- Smith, J.M., Stotts, D., and Kum, S.-U. (2000). An Orthogonal Taxonomy of Hyperlink Anchor Generation in Video Streams Using OvalTime, in *Proceedings of ACM Hypertext 2000*, ACM Press, pp. 11-18.