Hypervideo Summaries

Andreas Girgensohn, Frank Shipman, Lynn Wilcox
FX Palo Alto Laboratory, 3400 Hillview Avenue, Bldg. 4, Palo Alto, CA 94304

ABSTRACT

Hypervideo is a form of interactive video that allows users to follow links to other video. A simple form of hypervideo, called "detail-on-demand video," provides at most one link from one segment of video to another, supporting a single-button interaction. Detail-on-demand video is well suited for interactive video summaries, because the user can request a more detailed summary while watching the video. Users interact with the video is through a special hypervideo player that displays keyframes with labels indicating when a link is available. While detail-on-demand summaries can be manually authored, it is a time-consuming task. To address this issue, we developed an algorithm to automatically generate multi-level hypervideo summaries. The highest level of the summary consists of the most important clip from each take or scene in the video. At each subsequent level, more clips from each take or scene are added in order of their importance. We give one example in which a hypervideo summary is created for a linear training video. We also show how the algorithm can be modified to produce a hypervideo summary for home video.

Keywords: Hypervideo, video summarization, link generation, video editing

1. INTRODUCTION

To understand the content of a video, the user must view it linearly. Although there are tools for fast-forward and reverse, it is still time consuming to go through the entire video. A well-constructed video summary can help by providing an overview or outline of the content of the video. However, it is difficult to know how long the summary should be, and what information it should include. Most automatic summarization systems [2, 5, 7, 15] require users to specify the length of the summary, and include the most important information according to some pre-defined criteria. They do not address the issue that different people may have different information needs and may want different summaries.

We are exploring a summarization technique that uses interactive video to support viewers in watching a short summary of a video and in selecting addition detail-on-demand [11]. Our notion of detail-on-demand video has been influenced by interactive video that makes it possible for people viewing a video to make choices that impact what video they see. Examples of interactive video are DVDs that include optional side trips that the viewer can choose to take. For example, when playing The Matrix DVD with optional side trips turned on, the viewer sees a white rabbit icon in the upper left corner of the display when a link may be taken. These links take the viewer to video segments showing how the scene containing the link was filmed. After the side trip finishes playing, the original video continues from where the viewer left off.

In this paper, we use detail-on-demand video as a representation for an interactive video summary. Our approach generates a hypervideo composed of multiple video summary levels and navigational links between these levels. Viewers may interactively select the amount of detail they see, access more detailed summaries, and navigate to the entire video through the summary. Interaction with the video is through a special hypervideo player that displays keyframes with labels that indicate when a link is available, and what its content is. Unlike other keyframe-based summaries [17] or query-based access [8], the user interacts with video through the player rather than a separate interface. This allows video to be viewed on devices other than computers (e.g., DVD players).

Interactive hypervideo is a powerful format for user-specific summaries. However, authoring this type of summary is difficult. To address this difficulty, we developed an algorithm for generating hypervideo summaries automatically. These summaries are generated by first segmenting the video into clips and then selecting the clips for the different levels of the summary such that they summarize the video at the appropriate level. These clips must also be selected so

that there is continuity between summary levels, so the user does not get lost. Hyperlinks between related clips take viewers from less detailed to more detailed summary levels.

In the next section, we describe the detail-on-demand video model. This is followed by a description of the hypervideo player. We then describe an algorithm for automatically generating hypervideo summaries, and a system called Hyper-Hitchcock that supports editing hypervideo. We give two examples of hypervideo summaries, discuss previous work on video summarization, and present our conclusions.

2. DETAIL-ON-DEMAND VIDEO

Hypervideo allows viewers to navigate between chunks of video. General hypervideo allows multiple simultaneous links at any point in the video, for example, links from different actors on the screen to their biographies [4, 6, 13]. In this work, we use a simpler form of hypervideo called detail-on-demand video, where only one link is available at any given time. This representation provides a natural mechanism for user-specific video summaries [12]. The top-level video stream shows the highest-level summary, and links at various times provide the user with more information on a particular topic. Thus, the user can select the amount and type of content required.

2.1 Hierarchical Video with Links

In detail-on-demand video, each video sequence is represented as a hierarchy of video elements. Segments of source video are grouped into video composites, which may themselves be part of higher-level video composites. Links may exist between any two elements within these video sequences. The source element describes the source anchor for the link – the period of 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 end time.

Figure 1 shows links between two levels of video. The main level of video contains three clips. There is a link between clip1 and composite 1 as shown by the bold line. If the user follows this link, both clip 4 and clip 5 will be played in sequence since they have been grouped into a composite. There is also a link from clip 2 to clip 6 shown by the bold line. If the viewer takes the link during clip 2, only clip 6 in the second level will be played before returning to the main level.

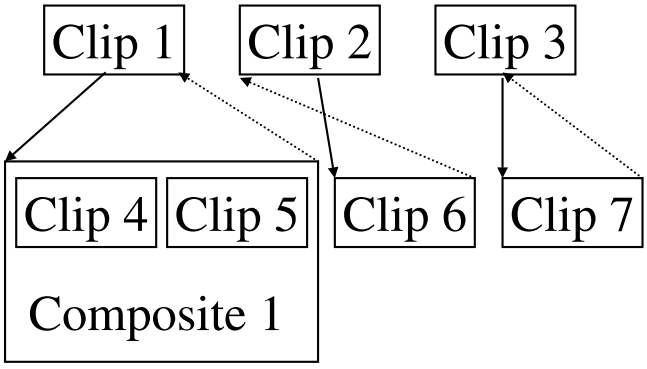

Figure 1. Hypervideo with composites and links. Solid lines show links and dotted lines illustrate link return behavior.

2.2 Link Labels and Behaviors

Links in detail-on-demand hypervideo have two characteristics that impact the resulting presentation. One is the link label that is shown to the user to indicate the content of the link. The other is the return behavior that specifies the player behavior when returning from the link. Link labels help users identify which links they want to follow. They can be added or modified using the Hyper-Hitchcock editor (see Section 5).

There are two independent link return behaviors that specify 1) what happens when the destination sequence of a video link finishes playing and 2) what happens when the viewer of the of the destination sequence ends playback before it finishes. Both have four options: 1) play from the point in the source video where the user took the link, 2) play from the

beginning of the source link, 3) play from the end of the source link, and 4) stop playback. Play from the end of the source link implies that playback begins at the next clip in the level.

The dotted lines in Figure 1 represent link return behaviors. For simplicity, we assume that these behaviors are the same for when playback ends and when the user returns early from the link. On return from composite 1, the playback begins at the end of clip 1 (i.e., the beginning of clip 2). This behavior is appropriate for links that provide more detail on the topic introduced in the source clip. After viewing the video link, the user does not need to see the original link again. On return from clip 6, playback begins at the beginning of the link source clip 2. This behavior is appropriate for links that provide pre-requisite material for the source clip. After viewing the link material, the user can better understand the original video. Finally, on return from clip 7, playback begins at the point the user took the link in the source clip 3. This is good for situations where the link provides alternate or related information.

3. HYPERVIDEO PLAYER

Viewing detail-on-demand video combines the characteristics of browsing the Web and changing channels on TV. As the viewer watches a video, the player indicates that a link is available from the currently playing clip by showing a keyframe and label for the link in the lower right hand corner of the application, as seen in Figure 2. The viewer can follow the link to see the link video or let the original video keep playing. The timeline of the player also indicates where links are available. It displays the labels for all available links in the currently playing video sequence to give the viewer an overview of the possible link destinations. In Figure 2 there are three link labels in the timeline, two of which are adjacent. In many cases the entire label does not fit in the timeline and only the first part can be shown. However, then whole label is shown as a popup when the user mouses over it.

Figure 2. The hypervideo player. Keyframes and labels to the right of the player indicate a link is available. Keyframes on the left of the player allow the user to return. Link labels are also displayed in the timeline.

The user follows a link by clicking on the keyframe or on an arrow button to the left of the timeline. When a link is followed, the video clip corresponding to the link begins to play. At the same time, another keyframe and label appear in the upper left hand corner of the application. This keyframe represents the video clip that the link came from, and allows the user to return to it. Link return behaviors are as specified in the previous section.

When a user takes a link or returns from a link, a brief video icon is played to provide an indication of what has happened. Video icons are short (about 2 seconds) video clips with distinctive audio tracks played between two clips of

video. These video icons were added because following hypervideo links happens quickly and can be confusing to users. The video icons make it clear that something is changing. Our current video icons are a video fading from gray to black with an audio track specific to the type of action. For example, an audio track with a rising pitch is played when a link is followed and another with a falling pitch when the user is returning from a link. We are also experimenting with animating the keyframes into the video to give the users more continuity.

4. AUTOMATIC HYPERVIDEO SUMMARIES

Detail-on-demand video is a good format for creating interactive hypervideo summaries. The summaries should be created so that an overview of the material is given at the higher levels, with links to more detail at the lower levels. Viewers can choose what parts of the video they want to see more detail on and follow links accordingly. The interactive player presents links and their labels, and aids the user in navigating the video.

However, manual authoring of hypervideo summaries is difficult and not cost effective, particularly if the video will only be used a few times. In this section, we describe an algorithm for automatically generating hypervideo summaries. This algorithm must specify the video clips, the composites, the links between them, and the link behaviors.

4.1 Segmenting Video into Clips

Hypervideo summaries are composed of video clips. If such summaries are to be generated from a linear video, the video first needs to be segmented into clips. In order to help with link generation, and similar to standard video analysis, we take a two-level approach to segmentation. At the top level, the video is segmented according to takes or scenes. These takes or scenes are then segmented into clips corresponding to shots.

If the original video is a produced video, for example, a training video distributed on CD ROM, the top-level segmentation is by scene, and the next level is by shot. We use the standard definition of these terms, namely that a shot is an unbroken sequence of video from a single camera, and that a scene is a sequence of related shots. Shots and scenes are segmented using standard algorithms [14, 18]. Typically, shot detection is performed first using inter-frame differences, and scenes are identified by grouping similar shots. Shot transitions, such as fades and wipes, are not included if the shot is used alone. However, when two adjacent shots are used in the hypervideo, the transition is inserted.

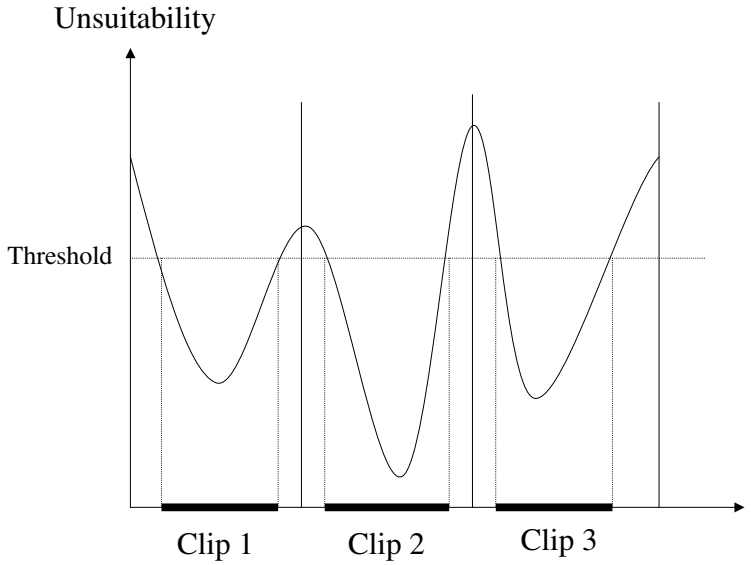

Figure 3. Unsuitability score for a home video Peaks in the score determine clips, and a threshold determines clip boundaries.

If the original video is a home video, for example, a video taken on a vacation or dance review, the first level of segmentation is based on takes, that is, times between camera on/off. Takes are detected automatically using information in the DV format. The second level of segmentation for home video relies on the notion of suitable video. With home

video, there is a lot of bad footage, for example, where the video is too dark or there is too much camera motion, which is unsuitable. We select shots by detecting segments of suitable video.

To do this, we first analyze the video and compute a suitability score based on brightness and motion [3]. Peaks in the inverse of the suitability score (the unsuitability score) are used to segment the video into clips for use in authoring. Figure 3 shows the unsuitability score for a take in a home video. Clips are determined based on peaks in the unsuitability score. In this case, there are three clips. The clip boundaries are determined by thresholding the unsuitability score. The threshold is set so that each segment contains at least 3 seconds of suitable video. Regions of high unsuitability between clips are treated like shot transitions in produced video – they are only included when two consecutive clips are used. A score for each of the clips is computed based on their average suitability. In this case, clip 2 is the best followed by clip 3 and then clip 1.

4.2 Clip Selection

To generate hypervideo summaries, we must select a subset of clips to include in the different summary levels. The basic algorithm is to include a single clip from each take or scene in the highest-level summary, and include more clips at each subsequent level. Ideally, the most general or most important clips from each take will be included at the highest level. Without video semantics, it is not possible to accurately select the best clips. Thus we use different heuristics for the case of produced and home video.

For produced video, we assume that the importance of each shot in the scene is the same as its order in the scene. This is correct if the scene begins with a general description and then proceeds sequentially with more detail. For home video, we order the clips in the scene based on their suitability. This allows the viewer to see better quality video at the higher levels, and get more detail later. Other importance measures are also possible [16] but have not been tested here.

4.3 Hypervideo Generation

The top level of the hypervideo summary begins with the most important clips from each take or scene. At each subsequent level, more clips are included. Clips from the same take or scene are grouped into composites, so that they are all played when the link is followed. When adjacent clips are grouped into a composite, the video between the clip boundaries is included in the playback video. Including these transitions and unsuitable video segments between adjacent clips gives the user more continuity.

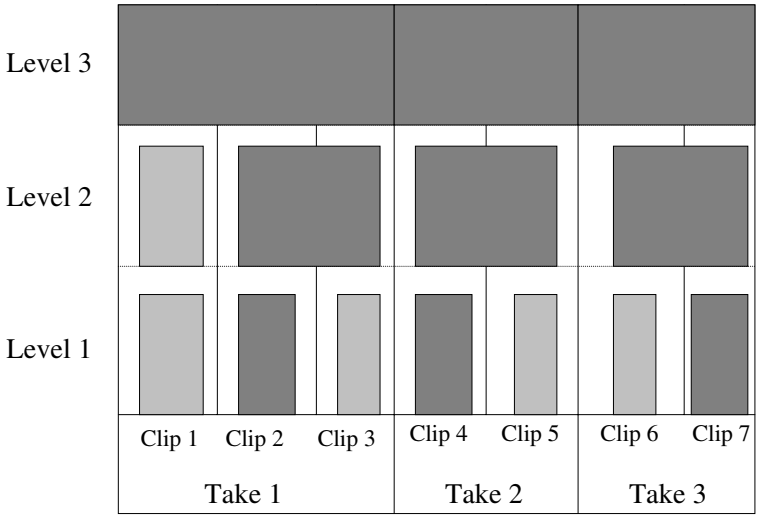

Figure 4. Clip selection for hypervideo summary of home video. Dark boxes indicate which video content is included at each level.

Links from higher levels to lower level composites are automatically generated. The return behavior depends on the type of video. For produced video, we assume that the lower level video provides a pre-requisite for understanding the higher level, and specify that the video playback begin at the beginning of the clip that the link came from. For home video, we

assume that the user will see more detailed content in lower level links and will not want to watch the higher level clip again, so we return playback to the end of the clip.

Figure 4 shows an example for a home video, where take 1 is from Figure 3. The top level includes the second clip in the take because it had the highest suitability score. The next level includes clip 2 and clip 3, and the video between them. The final level includes the entire take.

5. HYPER-HITCHCOCK

The algorithm described generates automatic hypervideo summaries with navigational links between them. In cases where the interactive summary will be used many times, such as in a professional training video, the authors can refine the automatically generated summary using Hyper-Hitchcock [12], an editing environment for detail-on-demand video. The system is also useful for adding labels to an automatically generated summary, or for changing link behaviors.

Figure 5 shows the Hyper-Hitchcock interface. It provides authors with a two-dimensional workspace to collect, organize, and link clips. The top left section groups source video clips into piles based on time or similarity. Authors drag clips from here into the workspace below where they can be ordered and grouped into composites. Links can be placed between any two clips or composites in the workspace. The lengths of clips and composites can be changed by resizing their keyframes or by moving handles in the timeline. Links can be labeled, and their return behavior can be specified.

Figure 5. Hyper-Hitchcock editor. The editor can be used to modify clips, links, and labels of an automatically generated hypervideo.

Authoring involves generating sequences of video clips. Given the limited space of the screen, it is convenient to create an iconic representation of a video composite. Hyper-Hitchcock represents the composite as a single image consisting of a collage of images from the individual clips. As an alternative representation, Hyper-Hitchcock provides a tree view of the hierarchy of composites and clips in the upper right section. Link colors and anchor positions in the keyframe indicate whether links originate and terminate at the whole composite or at components of the composite. Composites can be opened in a separate workspace to manipulate their components.

To support the use of the automatically generated hypervideo as the starting point for an authored summary, a graphical layout for editing the summary is automatically generated in the workspace. Each layer of the summary is presented as a horizontal list of clips and/or composites. Links are visualized as colored arrows in and out of keyframes. Figure 5 shows part of an automatically generated summary for a home video.

6. EXAMPLES

We tested our summarization algorithm on two videos. One was a video on do-it-yourself plumbing. The other was a home video covering a weekend family outing. Overall, summarization worked well, although with the produced video expectations on quality were higher so that the automatic summary was perceived as less effective than for the home video.

6.1 Produced Video Summary

Figure 2 shows the detail-on-demand video summary produced for the plumbing video. The video contained a number of sections separated by titles such as "Plumbing Basics," "Installing Shower Fixtures," and "Installing a Toilet." These sections were identified as scenes and created a good high-level structure for the hypervideo summary.

Each of these sections was broken into shots by the shot detection algorithm. In general, more shots were generated than would be desirable, since the same action, for example, replacing a pipe, was shown from different camera angles, and these were detected as shots. The result was that more levels than necessary were generated in the video. We used Hyper-Hitchcock to merge several of these segments, and to add link labels.

We then conducted a pilot study with a number of users. We asked them to answer specific questions about plumbing techniques and let them navigate the video to find the answers. In general, users were able to find the answers, and liked having the ability to jump around in the video. They liked having the keyframes on the left as a history of where they had been. They also relied heavily on the link labels as navigational aides.

6.2 Home Video Summary

Figure 5 shows the detail-on-demand video summary produced for a home video. In this case no further editing of the video was performed. The family liked interacting the video summary. The dad, who was more interested in his bicycle cross event, could quickly go to that part of the video and watch it in detail. The mom, who wanted to see video of the children, could watch those sections and skip the biking. In this type of video, the semantic structure is less important than the ability to navigate through the video.

7. RELATED WORK

Authoring interactive video is a relatively new area of research. While there are a number of hypervideo authoring tools, DVD authoring tools, and nonlinear video authoring tools, there is little work on automatically generating hypervideo. The closest areas of research are 1) the generation of linear video summaries, 2) hierarchy-based interfaces for accessing video, and 3) link generation for video.

7.1 Linear Video Summarization

One approach for providing more rapid access is to support skimming via shorter versions of the videos [2, 5, 7, 15]. Linear summaries of video are generated by a wide variety of applications. While the methods used to generate these summaries are of interest for generating individual levels of the hypervideo summary, these efforts do not include the generation of multiple summaries and the generation of links between these summaries. The key difference between linear and interactive multi-level summaries is that users can request additional detail for parts of the video rather than being restricted to a predetermined level of detail.

7.2 Hierarchic Interfaces to Video

Another approach is to support access to pieces of video via keyframes [17]. Video libraries let users query for pieces of video with particular metadata, e.g., topic, date, length [8]. 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 [9, 10]. Selecting a label or keyframe in a tree view is used to select a point for playback.

A 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 video representation of these tools does not include semantics beyond simple hierarchical composition. 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.

7.3 Automatic Linking of Video

There are a few hypervideo link generation algorithms that are loosely related to our approach. Most of these are aimed at generating links between video about common content as recognized by some (semi-)automatic means [4, 6]. OvalTine [13] tracks objects in video so that they can be used as link anchors for both manually and automatically created links. Algorithms for the automatic creation of links in video have focused on specific settings, such as news video [1, 18].

8. CONCLUSIONS

In this paper, we presented an approach for creating video summaries in the form of a multi-level hypervideo. These summaries are represented as detail-on-demand video, a form of hypervideo where video clips are hierarchically organized into video composites, links can exist between any two clips/composites, and only one link may be active at any given time. These automatically generated summaries can be modified in a direct manipulation editor called Hyper-Hitchcock. Such modifications range from adding meaningful labels to links to changing the lengths of video clips to changing the structure of the summary.

Our summarization algorithm produced good results both for produced linear training video and for home video. 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, detail-on-demand video 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. We are currently in the midst of a user study to determine the utility and usability of these summaries and of our hypervideo player. Early results are encouraging. Users are able to navigate such summaries can find information quickly.

REFERENCES

1. Boissière, G. Automatic Creation of Hypervideo News Libraries for the World Wide Web. *Hypertext '98 Proceedings*, ACM, New York, 1998.
2. Christel, M.G., Smith, M.A., Taylor, C.R., and Winkler, D.B. Evolving Video Skims into Useful Multimedia Abstractions. *Proceedings of CHI'98*, ACM Press, pp. 171-178, 1998.
3. Girgensohn, A., Boreczky, J., Chiu, P., Doherty, J., Foote, J., Golovchinsky, G., Uchihashi, S., and Wilcox, L. A Semi-Automatic Approach to Home Video Editing. *Proceedings of UIST '00*, ACM Press, pp. 81-89, 2000.
4. Grigoras, R., Charvillat, V. and Douze, M. Optimizing Hypervideo Navigation Using a Markov Decision Process Approach, in *Proceedings of ACM Multimedia*, ACM Press, pp. 39-48, 2002.
5. He, L., Sanocki, E., Gupta, A., and Grudin, J. Auto-Summarization of Audio-Video Presentations, in *Proceedings of ACM Multimedia*, ACM Press, pp. 489-498, 1999.
6. Hirata, K., Hara, Y., Takano, H., and Kawasaki, S. Content-oriented Integration in Hypermedia Systems, *Hypertext '96 Proceedings*, ACM, New York, pp. 11-21, 1996.
7. Lienhart, R. Dynamic Video Summarization of Home Video, *SPIE 3972: Storage and Retrieval for Media Databases 2000*, pp. 378-389, 2000.
8. Marchionini, G., and Geisler, G. The Open Video Digital Library, *D-Lib Magazine*. Vol. 8, No. 12, http://www.dlib.org/dlib/december02/marchionini/ 12marchionini.html, 2002.
9. Myers, B., Casares, J., Stevens, S., Dabbish, L., Yocum, D. and Corbett, A. 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, 2001.
10. Rui, Y., Huang, T.S., and Mehrotra, S. Exploring Video Structure Beyond the Shots. *International Conference on Multimedia Computing and Systems*, pp. 237-240, 1998.

11. Shipman, F., Girgensohn, A., and Wilcox, L. Creating Navigable Multi-Level Video Summaries. In *IEEE International Conference on Multimedia Computing and Expo*, vol. II, pp. 753-756, 2003.
12. Shipman, F., Girgensohn, A., and Wilcox, L. Hyper-Hitchcock: Towards the Easy Authoring of Interactive Video. In *Human-Computer Interaction INTERACT '03*, IOS Press, 2003 (to appear).
13. Smith, J.M., Stotts, D., and Kum, S.-U. An Orthogonal Taxonomy of Hyperlink Anchor Generation in Video Streams Using OvalTine, *Proceedings of ACM Hypertext 2000*, pp. 11-18, 2000.
14. Sundaram, H. and Chang, S.-F. Determining Computable Scenes in Films and their Structures using Audio-Visual Memory Models, in *Proceedings of ACM Multimedia*, ACM Press, pp. 95-104, 2000.
15. Sundaram, H. and Chang, S.-F. Condensing Computable Scenes Using Visual Complexity and Film Syntax Analysis. *Proceedings of ICME 2001*, pp. 389-392, 2001.
16. Uchihashi, S., Foote, J., Girgensohn, A., and Boreczky, J. Video Manga: Generating Semantically Meaningful Video Summaries, in *Proceedings of ACM Multimedia*, ACM Press, pp. 383-392, 1999.
17. Yeung, M.M. and Yeo, B.-L. Video Visualization for Compact Presentation and Fast Browsing, *IEEE Transactions on Circuits and Systems for Video Technology*. Vol. 7, no. 5, 1997.
18. Zhang, H.J., Tan, S.Y., Smoliar, S.W., and Yihong, G. Automatic Parsing and Indexing of News Video, *Multimedia Systems*, 2 (6), pp. 256-266, 1995.