Hyper-Interactive Video Browsing by a Remote Controller and Hand Gestures

Hui-Huang Hsu, Timothy K. Shih, Han-Bin Chang,
Yi-Chun Liao, and Chia-Tong Tang

Multimedia Information Network Lab,
Department of Computer Science and Information Engineering,
Tamkang University, Taiwan, R.O.C.
{hhsu, tshih}@cs.tku.edu.tw

Abstract. Interactive video browsing tools are designed for e-learning applications on future interactive TVs. The integrated system includes an authoring tool that produces multi-paths videos and a playback tool that uses video tracking technology and a remote controller. The playback tool enables multi-modal interaction between the user and a multi-story video clip. Three types of hyper-interactive controls are incorporated, which include a reference link of a video object to show supplementary information on the Web, a hyper link to enable hyper-video jumps, and a choice link for online answers to pre-designed questions. The underlying video is coded using the standard MPEG technology, with navigation information hidden in the user-defined data of MPEG. Thus, the default sequence of a hypervideo can also be presented using an ordinary video player.

1 Introduction

Interactive TV aims at giving full interactivity between the TV programs and the viewers. The viewers will be able to select needed video in a simple and easy way. The interface design and related issues has been under massive research and discussions in recent years [1, 2, 3]. On the other hand, hypervideo and multi-story video are also an important video technology under development [4, 5, 6, 7]. Video not only can be played in a linear sequence, but also can have multiple choices at certain points in the video. The story of a movie can progress in different ways and have different endings chosen by the viewer. An object in the video can be annotated with extra information in the form of a text file, an image, another clip of video, or a Web page. As long as the viewer triggers the reference link associated with it, the information will reveal. It would be very useful to integrate the hypervideo technology into interactive TVs to further enhance the interactivity.

How the reference link in a video can be triggered is an interesting issue. It is quite natural to do it by clicking a mouse, just like what is usually done on the World Wide Web. However, a mouse seems not well suited in the living room. People are more used to play with the buttons on a remote controller. In this paper, we propose multi-modal interaction for the playback tool. Besides using a remote controller, hand gesture of the viewer captured by a CCD camera can also be used for controlling hyper-video progress. The idea is shown in Fig. 1.

T. Enokido et al. (Eds.): EUC Workshops 2005, LNCS 3823, pp. 547 – 555, 2005.

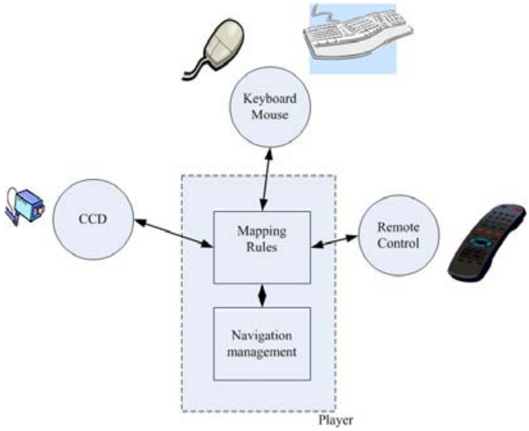


Fig. 1. Multi-modal hyper-interactive video viewing

For an e-learning application in the form of video compact disc, the learner will not need a computer to view the hypervideo content. A DVD player and a TV, which are standard appliances in the living room of a normal family, would be sufficient for the learner to start learning. Personalized learning content can be retrieved through the hypervideo structure of the learning content. For example, the learner can choose to or not to view a video clip with detail explanation of a vocabulary in a conversation for a language-learning scenario. Computers will not be needed.

In the following sections, we will introduce the authoring tool and the player of this integrated system. The multi-modal interaction and enabling technologies will then be delineated. And a brief conclusion will be drawn.

2 Hyper-Interactive Video Content Authoring

In order to produce hyper-interactive video content, an authoring tool is developed to divide the original video raw file into video clips. A directed-graph (digraph) structure composed of the video clips can then be constructed. One video clip can be used more than once in the digraph structure. The content producer can use the authoring tool with a user-friendly interface to produce the clips by simply marking in the starting time and marking out the ending time. Actually, the video file is not edited in any way. It is the marked points that are saved.

Fig. 2 shows the appearance of our authoring tool. There are several windows in this tool, including the video window, a digraph structure of marked hypervideo, and the metadata that can be added into the video.

The basic element of hypervideo is a video clip. A hypervideo is composed of several video clips. In our digraph-structured hypervideo, every node represents a video clip marked by the producer. The root node is the starting point of the hypervideo and is marked with a red square in the authoring tool. The red line between two nodes represents a video hyperlink between the two video clips. The audience can activate the hyperlink in a specified temporal-spatial domain to jump from one video clip to another. Nodes can have more than one branch. So the user can decide which video clip is the target via the multi-modal interaction. One of the hyperlinks of each node is set as the default link. If the audience does not make any choice, the video playing will proceed to the video clip with the default link.

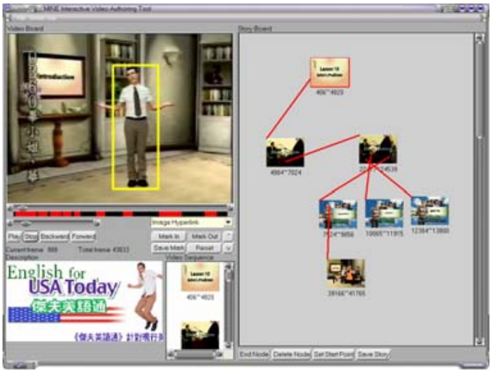


Fig. 2. The authoring tool of interactive-hyper video

Fig. 3 shows the result of a hypervideo structure. The producer can change the linearly played video sequence into a digraph-structured hypervideo. The audience can select the video clip he/she wants to watch by triggering a reference link. If they do not like the selected clip, they can go back to the parent video clip to choose another video hyperlink. The producer can also put some extra information to describe certain objects in the video clip. By the authoring tool, the producer can add text descriptions, existing image files, webpage files or URLs on the Internet to give more information to the audience.

Video annotation is usually used to enhance the semantics of the video object in the research of MPEG-4, MPEG-7 and video retrieval [8]. It is a big challenge to decide which objects to be recognized, tracked, and annotated. In our work, we adopt a manual and intuitive way to reduce the complexity of authoring.

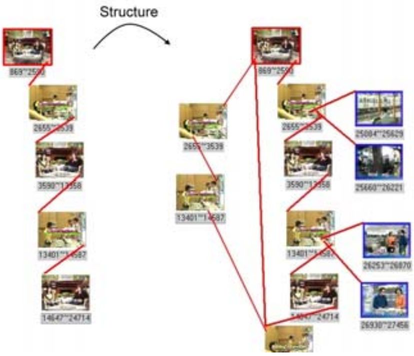


Fig. 3. A linear sequence versus a directed graph-like structure

3 The Hypervideo Player

In the proposed system, one key issue is how the annotated video can be played in a selected sequence exactly. A hypervideo presentation engine is designed for the hypervideo player. There are three components in the presentation engine: the navigation manager, the video decoder, and the video render (Fig. 4). They are described in details in the following.

- **Navigation_Manager:**
  The major component of the presentation engine is the navigation manager that lets the user browse the video sequence and receives the multi-modal interaction signal from the user. It can also control the process of video decoding when the user jumps to the next and previous video clips.
- **Video_Decoder:**
  This component is responsible for decoding the video signal like an MPEG decoder. The video decoder installed in the Windows OS is used directly.
- **Video_Render:**
  This component is used to render a video that comes from the output of the navigation manager and the video decoder. If one of the inputs to the video render is interrupted, the video render will output the default video sequence without using the hypervideo function.

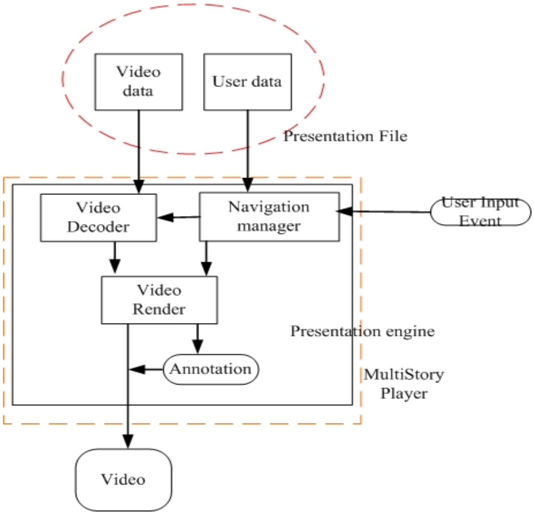


Fig. 4. The architecture of the hypervideo player

The video produced by the hypervideo authoring tool can also be played in other video players. Other players will play the default sequence of the digraph structure. The annotation added by the authoring tool will be ignored.

4 Multi-modal Interaction

Mouse clicking is natural to computer users on the WWW environment. Hypervideo is an extension of the idea of hypertext in Web pages to video. But here we look for other ways of interactions for future interactive TV applications. Under the scenario, people might not be used to mouse clicking. Thus, two other modes of interactions are introduced: 1. interaction with a remote controller, 2. interaction with camera-captured hand gestures.

4.1 Interaction with a Remote Controller

The first recommended device is a remote controller for TV. Most people are used to pressing buttons of a remote controller. However, current design of a TV remote controller needs to be enhanced to incorporate certain functions of the hypervideo player. A few buttons are redefined for such a purpose. The workflow of the remote control is shown in Fig. 5. In order to simulate the TV on a computer screen, an IR (infrared

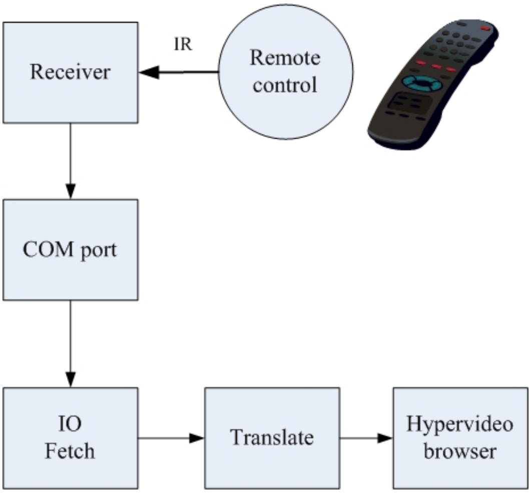


Fig. 5. The workflow of the remote control for the hypervideo player

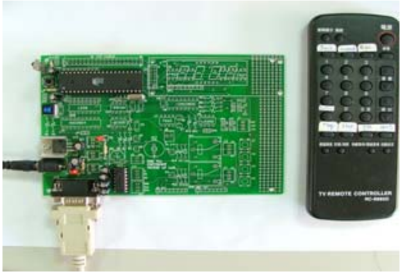


Fig. 6. The IR receiver and the remote controller

rays) receiver is added to the COM port to receive the IR signal from the remote controller. The IO fetch component receives data from the COM port and translates the data for the hypervideo player. Fig. 6 shows the IR receiver and the redefined remote controller.

Fig. 7 shows the situation that a viewer uses a remote controller to browse a hypervideo. The hyperlinks are shown in the video window with numbers. The viewer can activate the text, image, Web page, and/or video hyperlinks associated with the video clips by simply pressing the number buttons. Besides the hyperlinks, the viewer can also switch between play and pause or jump to a certain video clip.

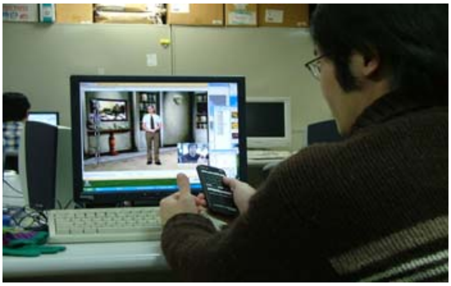


Fig. 7. Using a remote controller to browse hypervideo

4.2 Interaction with Camera-Captured Hand Gestures

A CCD camera is used as the second input device for the viewer to interact with the hypervideo browser. Hand gestures are used to replace mouse events. When the viewer waves his/her hand, the system finds the center of the palm and it is viewed as the mouse cursor. The viewer moves the hand to move the cursor to a certain hyperlink. When the palm is folded into a fist, the hyperlink is triggered. A low level CCD is sufficient to capture the user's gestures. Such a device can be acquired easily at a low cost.

Gestures are used as the event to trigger a hyperlink. The first step of recognizing gestures is to separate the hand in each frame. The second step is to find the center of gravity of the palm as the location of the mouse cursor. And the third step is to detect the palm is folding or unfolding [9, 10, 11].

The following procedures are used to separate the hand from the background:

1. Compute the difference of lightness in each pixel between the current frame and the former frame.
2. If the result is greater than the threshold, the value is set to 255. Otherwise, the value is set to 0.

This can be expressed by the following equation.

$$P_j(x, y) = \begin{cases} 1 & \text{if } |I_j(x, y) - B_j(x, y)| > T \\ 0 & \text{Otherwise} \end{cases}$$

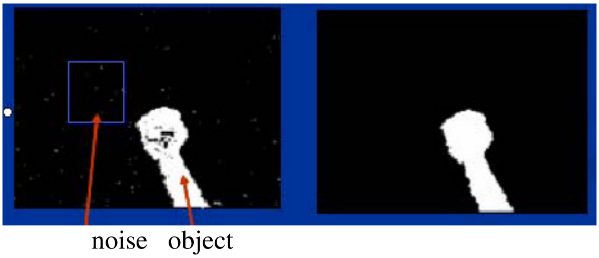
With an adequate threshold, we can get results of separating the foreground object from a still background.

Next we need to reduce the noise shown with the foreground object. The median filter and the closing operation are used. Here, the two techniques are briefly introduced.

Median filter: All the pixels values (in gray-level) in an N*N mask are arranged in a sequential order (from the smallest to the largest), then the middle value is selected from the ordered set to replace the value of the central pixel.

Closing operation: The closing operation performs the dilation followed by the erosion operation. Usually, it is used to fill in small holes or gaps and connect object's fragments.

By using these techniques, we can remove the redundant noise and get the hand area more precisely. Fig. 8 shows the hand area after removing noise.



(a) (b)

Fig. 8. Result of noise reduction (a) before and (b) after the noise reduction

Illumination change and background change are the two factors for getting an inaccurate background image. Without an accurate background, the foreground object cannot be separated. So, a dynamic background update mechanism is necessary. The major component of background adaptation is to calculate the difference between the current frame and background image. If the difference exceeds a threshold, the background image must be updated.

In this system, we use the number of on-off times of each hand object to determine if it is folding or unfolding. The procedure of determining on-off times is as follow:

1. Scan the source image horizontally from the top to the bottom with an interval of two pixels until whole image is scanned.
2. In the scanning process, the system checks the value change of neighboring pixels.
3. If there is a change from black to white, it is an "on."
4. If there is a change from white to black, it is an "off."
5. Sum up the total number of on's and off's.

With an adequate threshold, we can use the on-off number to determine whether the palm is folding or unfolding. Fig. 9 shows the two gesture configurations.

In our system, hand gestures are used to trigger the hyperlink in a hypervideo. The center of the palm is considered as the cursor to locate the target hyperlink. After the above-mentioned procedure, we can easily get the area of the hand. Then we compute the center of gravity of the palm to get a coordinate and use this position to select a hyperlink. Fig. 10 shows a locating result using the center of gravity of palm.

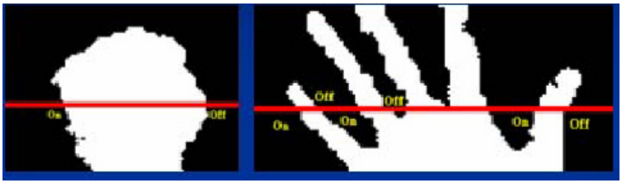


Fig. 9. Gestures: folding and unfolding

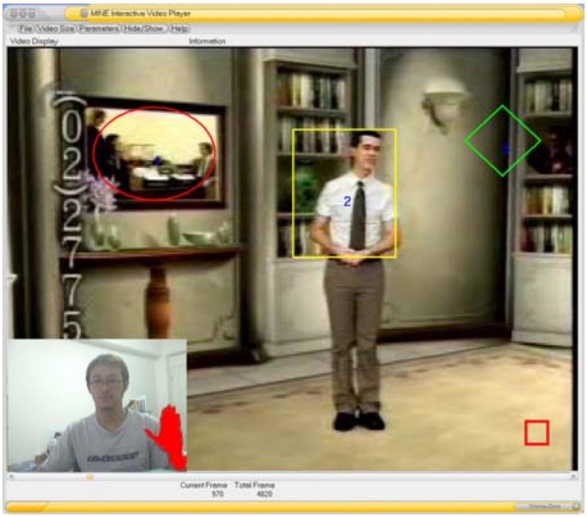


Fig. 10. Mouse moving and clicking by gestures

When the viewer moves his/her hand, the captured frame is analyzed and a coordinate is returned to the hypervideo player. The hypervideo render engine renders a small block indicating the location the palm points to. When the viewer folds the palm, the hypervideo player detects a low on-off number. This action is treated as the event of mouse clicking. So if there is a hyperlink in the area, it will be activated by the action.

5 Conclusions

In this research we integrated the hypervideo structure into the interactive TV framework. With the developed multi-modal interaction tools, the viewer can sit back and relax on a couch to learn an e-learning application with hypervideo content and enjoy the interactivity easily. The interactivity of the interactive TV is enhanced with the hypervideo.

Acknowledgement

This work was partially supported by National Science Council, R.O.C. under grant number NSC 94-2213-E-032-018.

References

1. Bing J., Dubreuil J., Espanol J., Julia L., Lee M., Loyer M., Serghine M., "MiTV: rethinking interactive TV," in *Proceedings Seventh International Conference on Virtual Systems and Multimedia* Oct. 25-27, 2001
2. Liang-Jie Zhang, Jen-Yao Chung, Lurng-Kuo Liu, Lipscomb J.S., Qun Zhou, "An integrated live interactive content insertion system for digital TV commerce," in *Proceedings Fourth International Symposium on Multimedia Software Engineering*, Dec. 11-13, 2002.
3. Cesar P., Vierinen J., Vuorimaa P., "Open graphical framework for interactive TV," in *Proceedings Fifth International Symposium on Multimedia Software Engineering*, Dec. 10-12, 2003.
4. Chang, H.-B., H.-H. Hsu, Y.-C. Liao, T. K. Shih, and C.-T Tang, "An Object-Based HyperVideo Authoring System," in *CD-ROM Proceedings of the Int'l Conf. on Multimedia Expo*, June 28-30, 2004.
5. Yoshiaki Hada, Hiroaki Ogata, and Yoneo Yano, "XML-based Video Annotation system for Language Learning Environment," in *Proceedings of the Second International Conference on Web Information Systems Engineering*, Vol. 1, Dec. 3-6, 2001.
6. Correia, P.L. and Pereira, F, "Objective evaluation of video segmentation quality," in *IEEE Transactions on Image Processing*, Vol. 12, Issue 2, Feb. 2003.
7. Nitin Sawhney, David Balcom, and Ian Smith, "Authoring and navigating video in space and time," in *IEEE Multimedia*, Volume 4, Issue 4, Oct.-Dec. 1997.
8. Jim Taylor, *DVD Demystified* (2nd edition), McGraw-Hill Professional, Dec. 2000.
9. Oka, K., Sato, Y., and Koike, H., "Real-time tracking of multiple fingertips and gesture recognition for augmented desk interface systems," in *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, May 20-21, 2002
10. Yang Liu and Yunde Jia, "A robust hand tracking and gesture recognition method for wearablevisual interfaces and its applications," in *Proceedings of the Third International Conference on Image and Graphics*, Dec. 18-20, 2004.
11. Dias, J.M.S., Nande, P., Barata, N., and Correia, A., "OGRE - open gestures recognition engine," *Proceedings of the 17th Brazilian Symposium on Computer Graphics and Image Processing*, Oct. 17-20, 2004.