HTML Paper

IMPLEMENTATION OF A REALTIME OBJECT-BASED VIRTUAL MEETING SYSTEM

Chia-Wen Lin[*], Yao-Jen Chang[**], Yung-Chang Chen[**], and Ming-Ting Sun[+]
[*]Department of Computer Science and Information Engineering,
National Chung Cheng University, Chiayi, Taiwan, R.O.C.

[**]Department of Electrical Engineering
National Tsing Hua University, Hsinchu, Taiwan R.O.C.

[+]Department of Electrical Engineering
University of Washington, Seattle, Washington, USA

ABSTRACT

This paper presents an H.323 standard compliant video conferencing system implementation. The proposed system not only serves as an MCU (Multipoint Control Unit) for multipoint connection but also provides a gateway function between the H.323 LAN (Local Area Network) and the H.324 WAN (Wide Area Network) users. The proposed video conferencing system provides user-friendly object compositing and manipulation features including 2-D video object scaling, re-positioning, rotating, and dynamic bit-allocation in a 3-D virtual environment. A segmentation scheme based on pre-stored background information is proposed for real-time segmentation of the foreground video objects at the client side. Chroma-key insertion is used to facilitate video objects extraction and manipulation. We have implemented the virtual conference system prototype with an integrated graphic user interface to demonstrate the feasibility of the proposed methods.

1. INTRODUCTION

With the rapid growth on multimedia signal processing and communication, virtual meeting technologies are becoming possible. A virtual meeting environment provides the remote collaborators advanced human-to-computer or even human-to-human interfaces so that scientists, engineers, and businessmen can work and conduct business with each other as if they were working face-to-face in the same environment. The key technologies of virtual meeting can also be used in many applications such as telepresence, remote collaboration, distance learning, electronic commerce, entertainment, Internet gaming, etc.

A virtual conferencing prototype, Personal Presence System (PPS), which provides user presentation control (e.g., scaling, repositioning, etc.) was firstly proposed in [1,2]. Due to the large computational demand, a powerful dedicated hardware is required for supporting the virtual meeting functionality, thereby leading to a high implementation cost. Recently several works have been done with a focus on the development of avatar-based virtual conferencing systems. For example, a virtual chat room application, V-Chat [3], has been developed to provide a 3-D environment with 2-D cartoon-like characters, which is capable of sending text messages and performing some predefined actions. While the 2-D avatar of V-Chat provides acceptable representation, several research works have been conducted on seeking for 3-D avatar solutions such as the Virtual Space Teleconferencing System proposed by Ohya *et al*. [4], the Virtual Life Network (VLNet) with life-like virtual humans proposed by Thalmann *et al*. [5], and the Networked Intelligent Collaborative Environment (NetICE) with an immersive visual and aural environment and speech-driven avatars proposed by Chen *et al*. [6]. Although the synthetic avatar-based solutions usually consume a small bandwidth, they may not provide satisfactory viewing quality due to the immature technologies to date.

ITU-T H.323 [7] is the most widely adopted standard to provide audio-visual communications over LANs. H.323 can be used in any packet-switched network, regardless of the ultimate physical layer. H.323 includes many mandatory or optional component standards and protocols such as audio codec (G.711/G.722/G.723.1/G.728/G.729), video codec (H.261/263), data conferencing protocol (T.120), call signaling, media packet formatting and synchronization protocol (H.225.0), and system control protocol (H.245) which defines a message syntax and a set of protocols to exchange multimedia messages.

In this paper, we address the implementation of a low-cost virtual meeting system which fully conforms to the H.323 standard. We propose a Personal Presence Multipoint Control Unit (PPMCU) which adopts the H.263 [8] video coding standard. The proposed virtual meeting system involves 2-D natural video objects and 3-D synthetic environment. We use chroma-key-based object extraction and manipulation schemes so that the developed techniques can be adopted in H.263 compatible systems. The concept can also be easily extended to MPEG-4 based systems.

The rest of this paper is organized as follows. In Section 2, we discuss the architecture of the proposed video conferencing system. Section 3 describes a pre-stored-background based video-object segmentation scheme for real-time segmenting out the conferees in a video conferencing session. Section 4 presents the proposed real-time implementation of the virtual meeting system. Finally, conclusions are given in Section 5.

2. PROPOSED SYSTEM ARCHITECTURE

The proposed PPMCU is a PC-based prototype which performs the multimedia communications and conversion and protocol translation among multiple LAN and WAN terminals. As shown in Fig. 1, the proposed PPMCU not only serves as a Multipoint Control Unit but also plays the role of a gateway to interwork between the H.323 (for LAN) and H.324 (for WAN) client terminals. The client side is basically an H.323 or H.324/I video terminal, which generates an H.323 or H.324/I-compliant bit-stream with integrated audio, video, and data content. The PPMCU server receives and terminates bit-

0-7695-1198-8/01/$10.00 (C) 2001 IEEE

streams from the H.323 and H.324/I client terminals as well as performs the Multipoint Controller (MC) and Multipoint Processor (MP) functions [7]. The MC and MP are used for protocol conversion and bandwidth adaptation respectively. In fact, the PPMCU server also includes the complete functions of the H.323/H.324/I client-side terminals.

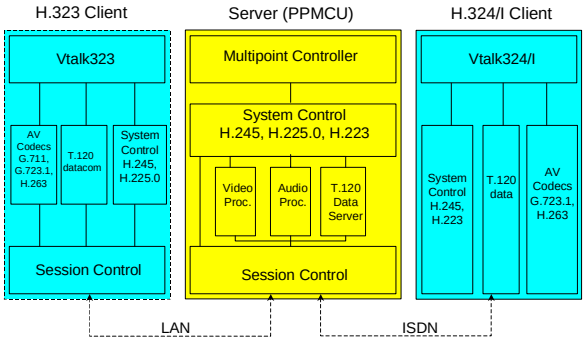

Fig. 1. The proposed PPMCU architecture.

Fig. 2. A virtual meeting example with 2-D video objects manipulated in a 3-D virtual environment.

Fig. 2 illustrates an example of virtual meeting which involves 2-D natural video objects (from the remote conferees) in a synthetic 3-D virtual environment. In order to make this possible, several key technologies need to be incorporated. For example, we need to be able to segment out the video objects in the video streams from remote locations and to compose them together so that these video objects appear interacting in the same environment.

Fig. 3 shows the proposed server-client architecture for the implementation of the H.263 compatible virtual conferencing prototype system. In a 3-D virtual environment, the location of each conferee is known, so are the relative positions of all the conferees. At the client side, the video object of each conferee is first segmented out. Then a chroma-key [9] is filled in the background. After chroma-key insertion, the client video accompanied with the 3-D position parameters are sent to the server. The server extracts all the client video objects according to the chroma-key information, transcodes the video objects to adapt to the available bandwidth and the user demands, and inserts the chroma-key again. Then the server sends back to each client the transcoded video objects filled with chroma-keyed background and their corresponding 3-D position parameters. The clients decode the received bit-stream, extract the chroma-keyed objects, then compose and render the 2-D video objects in a pre-stored 3-D synthetic environment according to the their 3-D position parameters.

In the process described above, object segmentation and manipulation are usually very computationally demanding. To meet the real-time requirement, as described in the following, we propose a fast object segmentation scheme which uses a pre-stored background information. The object compositing, manipulating, and rendering are performed in the video display cards which support OpenGL technologies, so as to reduce the computation burden at the client decoders drastically.

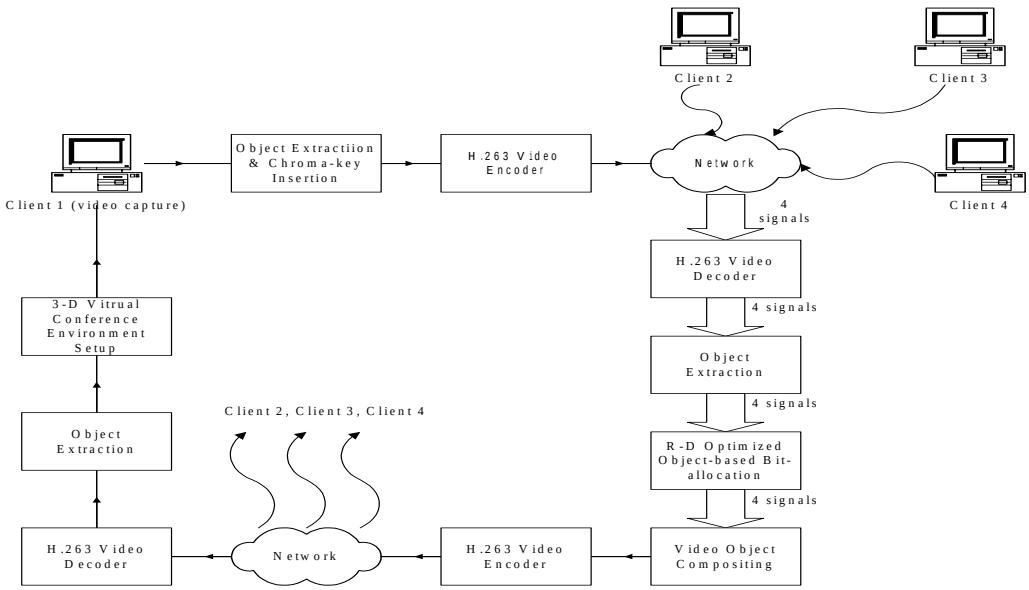

Fig. 3. The proposed server-client architecture for virtual meeting presentation

0-7695-1198-8/01/$10.00 (C) 2001 IEEE

3. REAL-TIME VIDEO OBJECT SEGMENTATION BASED ON STILL BACKGROUND SUBTRACTION

Extracting moving objects from a video sequence is a fundamental and crucial problem in many digital video applications, such as video surveillance, video editing, traffic monitoring and human extraction for virtual video conferencing or human-machine interface. Still background subtraction is an efficient method to discriminating moving objects from the still background [10]. The idea of background subtraction is to subtract the current image from the still background, which is acquired without before the objects move in. After subtraction, only non-stationary or new objects are left. This method is especially suitable for video conferencing applications, since in a video conference, the backgrounds for the conferees in general remain unchanged during the conference time. Should the background be changed, the users can capture and store the new background information again for object segmentation.

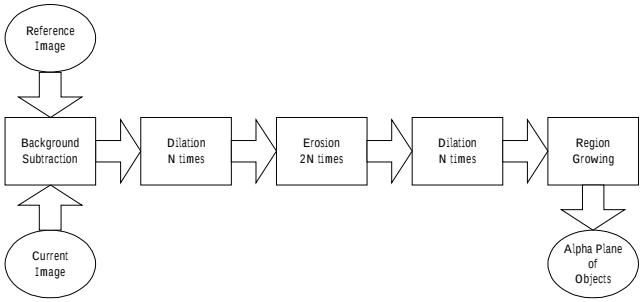

Fig. 4. The proposed object segmentation procedure

The proposed object segmentation procedure with still background subtraction is depicted in Fig. 4. In the background subtraction method, the background needs to be analyzed over several seconds of video. Each pixel value over a period of time may change due to camera noises and illumination fluctuations caused by light sources. The background scene is modeled by representing each pixel with two parameters, the mean and the standard deviation, during the training period. Each pixel in the current frame is first classified as either a background or a foreground class using the pre-calculated background model. The criterion to classify pixels is described as follows:

$$\textit{if } (|C(x) - \text{mean}(x)| > k \times \text{std}(x))$$
$$x \in \textit{ foreground}$$
$$\textit{else}$$
$$x \in \textit{ background}$$

where

$$\text{mean}(x) = \frac{1}{N}\sum_{n=1}^{N} R_n(x)$$

$$\text{std}(x) = \sqrt{\frac{1}{N}\sum_{n=1}^{N} R_n^2(c) - M^2(x)}$$

where x presents the index of pixels in the whole frame. $C(x)$ and $R_n(x)$ are luminance values of the pixel x in the current frame and the reference frame respectively. mean(x) and std(x) represent the mean and the standard deviation of the luminance values of the pixel x during the N reference frames. The statistics are updated frame by frame.

Background subtraction can roughly classify pixels of background and foreground, but the resulting segmentation result still may be noisy due to camera noises, illumination variations and inappropriate threshold selections. We propose to use the morphological filtering operations to remove small granular noises. Eight-Neighbor Dilation and Eight-Neighbor Erosion are used for suppressing noises in our system.

At the final step of object discrimination, a region glowing method, which connects neighboring foreground pixels as one region, is applied to the binary image. When there are more than one region available, the largest one is taken as the human object.

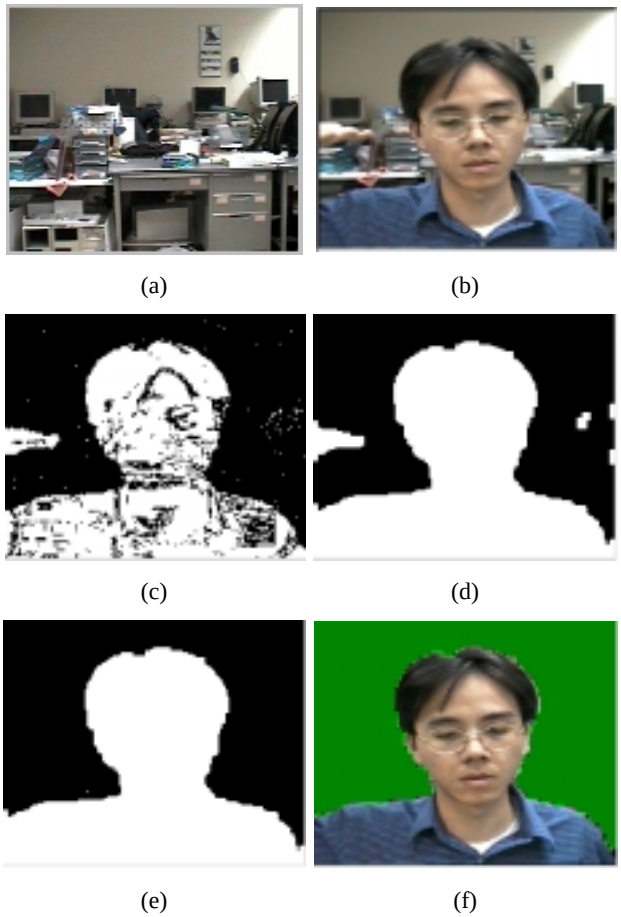

Fig. 5. The simulation result of the proposed object discrimination: (a) One of the background frames, (b) a frame containing a foreground object, (c) the alpha plane obtained by background subtraction, (d) the alpha plane obtained by morphological filtering, (e) the alpha plane obtained after the region growing method, (f) result of the object discrimination.

Fig. 5 shows the simulation result of the proposed object segmentation scheme. Figs. 5(a) and (b) show the pre-captured background and the image containing the foreground object respectively. Fig. 5(c) depicts the rough segmentations results after performing the still background subtraction scheme. The rough segmentation is still quite noisy. The result after applying the morphological filtering is illustrated in Fig. 5(d).

The small granular noises can be effectively eliminated using the morphological filtering process as shown. In Fig. 5(b), the noise with a larger area (a palm not belonging to the main object) on the left-side of the main video object was intentionally added, and it can be removed by the region glowing. From these results, we see that the proposed method can segment the video objects well for typical video conferencing image sequences in real-time.

4. VIDEO OBJECT COMPOSITING, MANIPULATION, AND 3-D SCENE RENDERING

As shown in Fig. 3, after extracting the video objects, the server transcodes the video objects using the dynamic bit-allocation method described in [11] and inserts a chroma-key background, then sends the chroma-keyed video back to the client terminals. The resulting video bit-stream still conforms to the H.263/H.263+ standard. The client side subsequently extracts the video objects from the received bit-stream and manipulates and renders the 2-D video objects against a 3-D virtual environment according to the 3-D location information of each conferee received from the server.

In a virtual videoconference, we need to place different persons at where we wish, against a 3-D environment which can even be artificial, allowing even overlaps between the people. We would also need to scale the segmented objects, so that a more realistic visualization of conferencing can be achieved.

Fig. 6. A snapshot of the proposed virtual meeting with natural 2-D objects in a synthetic 3-D environment using the OpenGL technologies.

By using 3-D computer graphics libraries such as OpenGL, a 3-D virtual environment can be created with 3-D object modeling parameters and rendered according to the specified viewing models. Integration with mouse operation functions, a 3-D immersive environment can be generated in which users can freely navigate and interact with other participants in the 3-D space. As shown in Fig. 6, we create a 3-D scene composed of a meeting room, wooden floor, and blue sky with clouds as the virtual environment. Using the OpenGL technologies, which have been supported by most of the commercial 3-D display cards, at the client side, the computational load is thus shared by the graphic chips on the display card, thereby most of the computing power of the client terminals can be dedicated to the object segmentation and video encoding and decoding. Furthermore, speech signals are mixed according to the virtual distance between the sender and the receiver in the 3-D space for providing an aural immersive virtual environment.

5. CONCLUSIONS

We have described the implementation of a video conference system which plays the role of both MCU and gateway. The LAN users through Ethernet and the WAN users through ISDN can be brought together via the proposed PPMCU. The proposed PPMCU provides an integrated platform for video, voice, and data communications, and is fully compatible to the H.323/H.324 standards. The proposed PPMCU provides advanced personal presence controllable object processing features such as scaling, re-positioning, rotating, and dynamic bit-allocation in real-time. We have presented efficient methods for implementing video object segmentation and the user-friendly object processing features. We have implemented a virtual conference system prototype to demonstrate the feasibility of the proposed methods.

6. REFERENCES

[1] M. E. Lukacs and D. G. Boyer, and M. Mills, "The personal presence system experimental research prototype," *IEEE Int. Conf. Comm.*, vol. 2, pp. 1112-1116, Jun. 1996.

[2] M. E. Lukacs and D. G. Boyer, "A universal broadband multipoint teleconferencing service for the 21st century," *IEEE Comm. Magazine*, vol. 33, no. 11, pp. 36-43, Nov. 1995.

[3] Microsoft V-Chat 2.0, available at http://www.microsoft.com/ie/chat/vchatmain.htm.

[4] J. Ohya, K. Kitamura, F. Kishino, N. Terashima, H. Takemura, and H. Ishii, "Virtual space teleconferencing: real time reproduction of 3D human images," *Journal of Visual Comm. Image Representation*, vol. 6, no. 1, pp. 1-25, Mar. 1995.

[5] T. K. Capin, L. S. Pandzic, N. Thalmann, D. Thalmann, "Virtual human representation and communication in VLNet," *IEEE Computer Graphics and Applications*, Vol.17, No.2, pp.42-53, 1997.

[6] W. H. Leung, K. Goudeaux, S. Panichpapiboon, S.-B. Wang and T. Chen, "Networked Intelligent Collaborative Environment (NetICE)," in *Proc. of 2000 IEEE Intl. Conf. Multimedia and Expo*, vol. 3, pp. 1645-1648, NY, USA, Jul. 2000.

[7] ITU-T Recommendation H.323, "Visual telephone systems and terminal equipment for local area networks which provide a non-guaranteed quality of service". 1998.

[8] ITU-T Recommendation H.263, "Video codec for low bit-rate communication". 1996.

[9] T. Chen, C. T. Swain, and B. G. Hsakell, " Coding of subregions for content-based scalable video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 256-260, Feb. 1997.

[10] I. Haritaoglu, D. Harwood, and L.S. Davis. "W4: Who? When? Where? What? A real-time system for detecting and tracking people," in *Proc. The third IEEE Intl. Conf. Automatic Face and Gesture Recognition*, pp. 222-227, Los Alamitos, CA, 1998.

[11] C.-W. Lin, T.-J. Liou, and Y.-C. Chen, "Dynamic rate control in multipoint video transcoding," *Proc. IEEE Int. Symp. on Circuits and System*, II.17~20, May 2000, Geneva, Switzerland.