DESIGN AND IMPLEMENTATION OF A WEB-BASED SURVEILLANCE SYSTEM USING INTERNET MULTICAST COMMUNICATIONS

Jan-Ming Ho[*], Ray-I Chang[*], Jie-Yong Juang[#], Chia-Hui Wang[#]
[*]Institute of Information Science 20, Academia Sinica
NanKang 115, Taipei, Taiwan
Tel:886-2-27883799, FAX:886-2-27824814, Email:{hoho, william}@iis.sinica.edu.tw
[#]Department of Computer and Information Science, National Taiwan University
#1 Roosevelt Rd. Sec. 4, Taipei, Taiwan 106
Tel:886-2-23625336, Fax:886-2-23628167, Email:{juang, d5526006}@csie.ntu.edu.tw

Abstract: *Multicast connections with the one-to-many communication model can reduce the loads of server and network by removing redundant traffic. In this paper, based on this architecture, we present our design of a surveillance system where users can retrieve live and pre-recorded surveillance video from any spots under surveillance through wire/wireless WEB devices or PSTN auto-dialup networks. To provide effective and efficient delivery of surveillance videos over Internet, mechanisms for application-level traffic shaping and motion detection are proposed. Implementation results show that our traffic shaping method helps this system remit the performance penalty from the low bandwidth network. Additionally, the motion can be detected with low false alarm in real-time. In this paper, we present not only the experiences of multimedia system implementation but also schemes to utilize multicast communications. It contributes to design and implementation of other applications in the future.*

KEYWORDS: *Multicast, Codec, Traffic Shaping, Motion Detection, MBone.*

INTRODUCTION

Traditional surveillance systems only provide analog services in hardware. Security guards must stay at security room and look at arrays of CCTV (Closed Circuit TeleVision) or play back the videotapes sequentially to find out the surveillance events. This demanding task is very inefficient. According to the current technologies of low-cost powerful PC (personal computer) and wire/wireless networks, a flexible application environment with automatic event detection and random video playback functions can be provided. In this paper, we introduce the design and implementation of a Web-based surveillance system using Internet multicast communications. It provides not only the low bit-rate digital data, but also the remote control capability. Users can

retrieve live or pre-recorded surveillance video through wire/wireless WEB devices or PSTN auto-dialup networks as shown in Fig. 1. To provide effective and efficient delivery of surveillance videos over Internet, mechanisms for application-level traffic shaping and motion detection are proposed. Implementation results show that our system can remit the performance penalty from the low bandwidth network and detect the surveillance events with low false alarm rate in real-time.

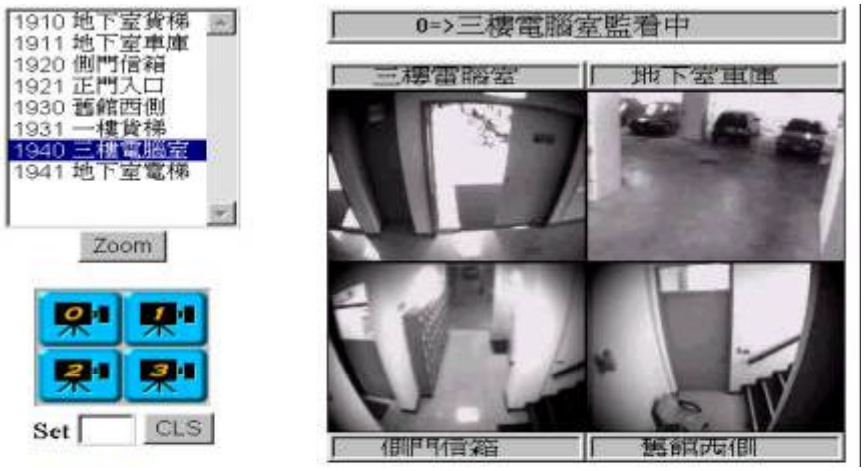

Fig. 1. An overview of the execution result of the proposed WEB-based surveillance system.

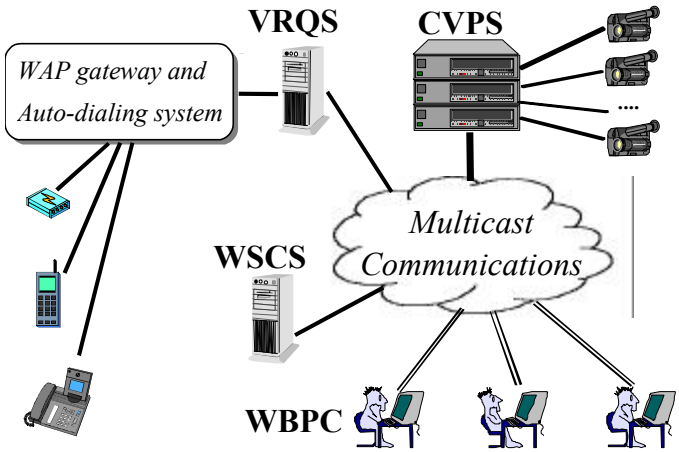

Fig. 2. An overview of the proposed architecture with additional WAP gateway and auto-dialing system (to security guard's pager or phone).

1. SYSTEM ARCHITECTURE

Our surveillance system can be divided into four subsystems: CVPS (compressed video pumping servers), WBPC (Web browser plug-in client), WSCS (Web server CGI scripts) and VRQS (Video Recording and Querying Server) as shown in Fig. 2. All these sub-systems are running on Microsoft Windows.

Compressed Video Pumping Servers (CVPS): The analog signal of each video capture card

comes from an immobile monitoring camera installed at a sensitive surveillance spot. CVPS gets the raw video data from the capture cards and call the Microsoft Windows VFW API to compress the input data into H.261 video frames. H.261 describes the video coding and decoding methods for the moving pictures component. While JPEG treats each picture independently, H.261 predicts the current picture from the previous one to reduce redundant information in a video stream. Moreover, unlike MPEG [12], H.261 provides encoders and decoders to operate symmetrically in real-time.

Web Browser Plug-in Client (WBPC): WBPC is a resident plug-in software module used in the browser to provide a user interface for selecting and displaying the surveillance video transmitted from networks. At first, WBPC tries to join the corresponding multicast groups for the surveillance spots that the user wants to watch over. When the video packets are received, this module decodes them via VFW API and displays them sequentially.

Web Server CGI Scripts (WSCS): WSCS is a resident program running at the Web server that receives the interactive commands from clients. To guarantee its reliability, all the links between WSCS and WBPC use connection-oriented protocol, i.e., TCP. Miscellaneous information, e.g., camera locations and camera identifiers (which map to the IP addresses of multicast groups), are put into a database in WSCS. WSCS forwards the respective attributes to WBPC from the built-in database upon the request of the command issued from remote users at WBPC.

Video Recording and Querying Server (VRQS): VRQS receives video packets from CVPS. However, it won't display the video frames but analyzes and saves them as files. Therefore, remote users can play and query the pre-recorded surveillance video through the Web page. It furnishes exciting features of a surveillance system to indicate and watch the surveillance videos at any time instance. (The reason why we install VRQS at another machine is to reduce the loading of CVPS. Usually, VRQS should run on a machine which is close to CVPS such that network channel between them is lightly loaded.)

2. APPLIED SCHEMES

2.1. Internet Multicast Communications

Different from video-conferencing that requires symmetric multimedia transmission, a surveillance system acts like a one-to-many sender-listener communication model. It is very suitable for implementing on a multicasting network. Multicast transmission schemes consider the sending of data from one to many recipients, or many to many recipients. Each class-D IP address used for multicast transmission indicates a multicast group. Without loss of generality, we consider a simplified model which contains only one CVPS and only one camera source to discuss the beneficial of multicast transmission of the proposed system. Assume that g different users have selected the same one surveillance spot. Our CVPS will contribute a B bps traffic load

to the network. We elaborate four different loading components that will impact the network performance. The first one is the load to the end-to-end network. The second one is the load to the network driver on WBPC. The third one is the load to the network driver on CVPS. The last one is the load to the other irrelevant network driver. It's known that only broadcasting connection will have load impact to the irrelevant machines on the same subnet. Then, we can estimate the loading components for different types of transmission schemes. As shown in Table 1, we can find that extensively scalable connections of multicasting perform less loading than the others do.

Table 1. The loading components for different types of transmission schemes. (M is the number of other irrelevant network-enable machines on the same subnet).

	Broadcast**	Unicast	Multicast
1. network load:	B	$g*B$	B
2. WBPC network driver load:	$g*B$	$g*B$	$g*B$
3. CVPS network load:	$2*B$	$g*B$	B
4. Irrelevant network driver load:	$M*B$	0	0
Total estimated load (1+2+3+4)	$\boldsymbol{(M+g+3)*B}$	$\boldsymbol{3g*B}$	$\boldsymbol{(g+2)*B}$

Note that, in the Internet, not all routers can support multicast transmission. A popular solution is to install MBone (Multicast Backbone) [4] tunnel between two network islands where the multicast packets can not be routed. The installed MBone machine on the edge of one network island will encapsulate multicast packets as normal IP packets and forward these encapsulated IP packets with predefined transmission rate to the other MBone machines installed on the edges of other network island via the virtual tunnel. Then, the MBone machine that receives the encapsulated IP packets will remove the headers of these IP packets to recover the original multicast packets. At last, the MBone machine delivers them to its own multicast-capable network island. The TTL (Time To Live) value in the packet header can be redefined by the applications. In a multicast packet, the TTL value will decrease by one when it visits a multicast router (including MBone tunnel). If the router finds out the TTL reaches to zero, it will stop forwarding the multicast packet. So the TTL value can enable the transmission scope for the Internet multicast applications.

2.2. Proposed Traffic Shaping (Spacing) Scheme

The non-discriminated sharing of network resource in Internet makes no guarantee of timely delivery between senders and receivers. The performance metrics such as delay, delay jitter, and packet loss rate should be considered for multimedia applications. In a surveillance system, as the lost of image frames means that it's not possible to indicate moving object or the intruder, packet loss (*i.e.* video frame lost) should be considered more important than the other metrics. In this paper, we based on a codec-level macro-block updating scheme proposed in [10] (to increase the codec robustness for the video transmission over the Internet) to revise and integrate the

previous work of application-level traffic shaping [2] mechanism. To reduce the buffering burstness in the path to WBPC, we not only divide the compressed image into small PDUs (Protocol Data Unit) but also modify the previous shaping method for the variation of the number of packets in frames. If the maximum frame rate of the video compressed in the system is N frames per second, the transmission of the S packets in a video frame after segmentation should be accomplished within $1/N$ second. To reduce the back-to-back burst for transmitting these P segmented packets, each packets of the video fame should be spaced out within $1/(P*N)$ second (called inter-burst interval). Considering the overhead of segmentation processing time and unpredictable packet delivering time to the network driver of Microsoft Windows -- a non-real time operating system, the inter-burst interval should be deducted by the processing time of the overheads above. The proposed traffic shaping (spacing) method is described as follows.

Algorithm: Traffic Shaping

/* F_i: {F_i | *the size of i th compressed frames; i= 1,2,3….}.*
N : the expected frame rate.
G : size of the segmented packets.
p : *variables for the segmented number of packets in a frame.*
t, u : *variables for system tick count in ms. */*

While (F_i is available){

Let $p = \lceil F_i / S \rceil$ /* take the integer ceil function of F_i/S */

Repeat {

t = getTickCount( ) /* Get the current system tick count t via system call. */

get one packet from p segmented frame, and send it out.

$u = 1/(p*N) - [$ getTickCount( ) $- t$]. /* $1/(p*N)$ is ideal inter-burst interval */

If (u >0) then Sleep(u). /* sleep for u ms (u is the time in real-life) */

$p=p$-1.

} **Until** (p is zero).

} /* While */

2.3. Proposed Motion Detection Scheme

Theoretically, the degree of motion definitely affects the size of intra-frame that contains the different information with the previous frame. However, in real world, the variation of brightness

(in which the human vision almost can not tell the difference) will also affect the size of frame. In this paper, to reduce the false alarm, we consider the signal of light as white noise in the video background. Based on this idea, a simple method is proposed to filter out the noise with variable average in energy and detect the motion. This algorithm provides an effect and efficient mechanism to the system to achieve relatively low false alarm without adding much loading. In our implementation, N represents the length of the cycle in which there is only one intra-frame (*i.e.* the 1st frame is intra-frame). The set $S_{j,k}$ is a silence period in which there is no motion and intruder occurs from the j-th frame to the k-th frame. The algorithm is briefly shown below.

Algorithm: Motion Detection

/* F_i is the size of i-th frames.

$S_{j,k}$: no motion object and intruder found in from the frame j to the frame k.

m: $m = \frac{1}{31}\sum_{i=2}^{32} F_i$, F_1 is intra-frame.

M: moving average of noise power at F_i of window size W, where frame i is inter-frame.

$T = m + P$ *where* P is the maximum value of (F_i - m) for i=2,3, ..., 32.

D : {D | $F_i > T$, for i= l, 2, …, D}. */

```
Find out P and m in S_{1,32.} /* make sure that no motion happens in S_{1,32} */
Calculate out T by sum of P and m.
d = 0.                          /*Set duration count d to zero */
While (Frame i is inter-frame; for i > 32){
   If ( F_i > T) then {         /* compare the size of inter-frame i with threshold T */
      d = d+1.                  /* increase the duration d count by 1 */
      If ((d > D) and (the alarm is not set)) set alarm
   }
   else{
      If (d > D) reset d to zero.     /* motion just leave from the monitoring video */
      Calculate the M by moving average method
      T = P + M                 /* reset the threshold T for the variable noise power */
   }
}
```

While the sizes of later inter-frames have been larger than the threshold T and last for a time period D at the count of frames where the threshold T is the sum of peak value P and m, this algorithm will consider that there is motion in the monitoring video and set alarm. Because the average of the noise power will vary from time to time, the moving average method [14] is used to keep track of the current average noise energy in windows size W in the silence period $S_{j,k}$. To reduce false alarms, the heuristic value of duration D must be long enough to validate the motion for the interfering noise. However, it must be short enough to reduce the miss ratio of the motion

in a flash. The best value of *D* will depends on the frame rate because the higher frame rate will have more frames with motion inside. We also will have to adaptively adjust the threshold *T* for the intensity of white noise will change from time to time. The best period to adjust the threshold is during the silence period where no motion happens in the live video. So the adaptive threshold *T* and the duration *D* are very important factors for the hit and miss ratio of motion detection. The evaluation result of the proposed motion-detection algorithm will be shown in the next section.

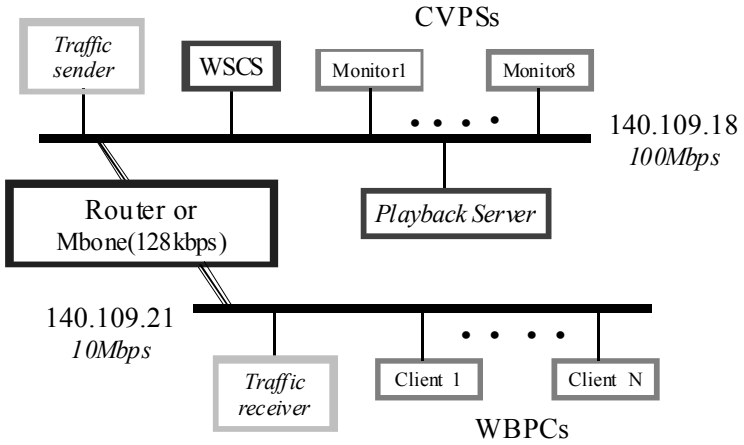

Fig. 3. The configuration of the experimental testbed.

Table 2. Testbed equipment list of CVPSs.

Operating Systems	CPU	NETWORK CARDS	VIDEO CAPTURE CARDS	CAMERA LOCATION	Average Receiving Rate
Windows 95	Pentium 120	10Mbps (Intel82557)	(a) BT848 Chip	B1Cargo Elevator	2.8
			(b) Zoran Chip	B1 Garage	2.8
	Pentium 133	10M bps (Etherlink III)	(a)	East Side Entrance	2.7
			(b)	Lobby	2.6
Windows NT 4.0	Pentium II 400	10/100M bps (Intel82557c)	WinNov System	West Side Entrance	4.5
				1F Cargo Elevator	4.3

3. PERFORMANCE

A series of experiments are conducted on the testbed as shown in Fig. 3 to investigate the performance of the proposed system. The equipment list of CVPSs is shown in Table 2. The video frame size is CIF in all kinds of our experiments. The network backbone of the totally 8 CVPSs is the Ether-switch of 100Mbps. In fact, there are two types of connection for remote WBPC users. One is from broadband Internet across one multicast router and the other is from the low bandwidth Internet, MBone (maximum bandwidth of the tunnel is up to 128kbps in our testbed). We examine the performance of traffic shaping method by executing all strategies on the connection of MBone. In the following paragraphs, we show some representative results for traffic shaping and motion detection methods, respectively.

Table 3. Performance Results for different traffic shaping strategies (6 streams in tunnel).

Surveillance Location	Burst Strategy			Strategy A			Strategy B		
	Send Rate	Receive Rate	Frame Loss	Send Rate	Receive Rate	Frame Loss	Send Rate	Receive Rate	Frame Loss
B1 cargo elevator	2.6	2.2	171	2.8	2.6	55	2.7	2.6	80
B1 garage	2.6	2.2	168	2.9	2.7	51	2.8	2.7	68
East side entrance	2.6	2.3	118	2.7	2.5	73	2.7	2.6	82
Lobby	2.6	2.2	169	2.8	2.7	144	2.8	2.7	125
West side entrance	4.4	3.6	546	5.1	4.9	144	5.2	4.9	207
1F cargo elevator	4.5	3.6	524	5.1	4.9	170	5.2	5.0	120

In our experiments for the new approach, we propose two testing strategies to validate the effect of the enhanced traffic shaping method. To maintain the 5 output frame rate (*i.e.* $N = 5$) of the H.261, in strategy A, each segmented video packet is of 512 bytes long (*i.e.* $S = 512$). In strategy B, each video packet is of 128 bytes long. In the modified traffic shaping method, the inter-burst interval [2] is adaptive (*i.e.* $200/p$) rather than constant and its value will depends on the processing time of segmentation overhead and the system function call for sending out the packet in Windows. Besides, the strategies with smaller PDU than strategy B was applied, the more overhead of the segmentation and the PDU headers the application will suffer. In the experiments, totally X surveillance spots are selected from remote WBPC users. That's to say X multicast streams will flow in the Mbone tunnel to the users. From our measurement, each surveillance spot will contribute more than 32kbps traffic. If X is larger than 4, the 128kbps MBone tunnel would not be able to sustain the overloaded traffic. Besides, we record the videos from a surveillance spot (*i.e.* Lobby) for 20 minutes. We playback the pre-recorded videos from VCR and feed the output video signals to the capture card in CVPS and fairly examine the performance at WBPC users for the cases when X equal to 6 (as shown in Table 3). We assume other vide streams (*i.e.* X-1 live streams) in the tunnel as the multicast UDP background traffic. Our unlisted experimental result for the broadband Internet shows that WBPC performs no packet loss and a little bit slow frame (decreased by 0.1 frame per second), even there is background traffic with over 8Mbps in the WBPC network with 10Mbps bandwidth when traffic shaping method is not applied yet. For the low bandwidth network, WBPC suffers that the information of motion object and intruder in the video may be lost due to a lot of packet loss. However, the VRQS module will provide an interface to review the intact pre-recorded surveillance videos through the Web page in the reliable file transfer protocol (*i.e.* FTP). After we apply both strategies we proposed in the traffic shaping method, the packet losses are all decreased without sacrificing the frame rate. Surely, we present again the performance of the modified traffic shaping method in which both shaping strategies can undertake the heavy loading in the network.

In playback function of the surveillance system, the surveillance videos are pre-recorded as video files. Our experiments will playback the video file and impose on the frame size

information in the video files to validate the motion detection algorithm. In our implementation, currently D is set to 2 in the implementation and the window size W of the moving average method is 8. The experimental results of the proposed adaptive motion detection algorithm shows low false alarm and low miss rate[1] (as shown in Table 4) in the ASIS surveillance system.

Table 4. Performance results for our motion detection scheme.

Surveillance Location	Set Alarms	False Alarms	Miss Alarms	Test Time
B1 cargo elevator	10	5	0	1700:1859
B1 garage	13	0	3	1700:1859
East side entrance	33	0	0	2100:2259
Lobby	32	0	1	2100:2259
West side entrance	0	0	0	1815:1859
1F cargo elevator	1	0	0	1815:1859

4. CONCLUSIONS AND FUTURE WORK

In this paper, we first describe the real-time multicast surveillance system implemented and operational in our institute. This design has the flexibility for a user to select among different surveillance spots. User can also choose to receive only the live video stream of interest for viewing them in zooming size. The one-to-many multicast model also presents the benefit of deploying multicast transmission for surveillance applications. By observing the packet loss of real-time live video data in MBone when the network bandwidth is low, the application-level traffic shaping mechanism controls the burst of sending the compressed video packets to the receivers. It reduces the burst arrival in the buffers along the path to the receivers and decreases the chance of overflow in the buffer to achieve low packet loss without affecting the frame rate. Though many others provide effective and efficient motion detection solutions in all kinds of monitoring application indoors and/or outdoors, most of them are furnished by fancy and expensive hardware or by non-standard software codecs. Our preliminary experiments on the H.261 video frames shows good results of low false alarm and low miss ratio for the low-complexity motion detection algorithm, but the ideal goal of motion detection is to achieve virtually zero false alarm and miss rate in all kinds of surveillance scenes. To explore and elaborate the low-complexity optimal noise eliminating function for virtually zero false alarms and miss rate in the inexpensive surveillance application is our goal in the near future.

REFERENCES

[1] C.H. Chang, M.C. Chen, J.M. Ho, M.T. Ko, K.H. Tsai, C.F. Wang, and D.W. Wang. ASIS-Academia Sinica multimedia interactive system. In Proceedings of the 1st Workshop on Real-time and Media Systems, pages 30-31, Taipei, Taiwan, July 1995.

[1] Though the surveillance spot in 3F computer room shows high miss alarms, we found that motions in the silence period will truly effect the miss rate for the threshold T.

[2] Y.S. Sun, C.F. Ku, Y.C. Pan, C.H. Wang, J.M. Ho. Performance Analysis of Application-Level Traffic Shaping in a Real-Time Multimedia Conferencing System on Ethernets. In Proceedings of the 21st IEEE Conference on Local Computer Networks (LCN '96).

[3] M.W. Li, C.F. Ku, Y.C. Pan, C.H. Wang, M.C. Chen, J.M. Ho, M.T. Ko. Real-time Distributed Clock Synchronization over Ethernet. In Proceedings of the 2nd Workshop on Real-time and Media Systems pages 19-26, Taipei, Taiwan, July 1996.

[4] Vinary Kumar , Mbone / Tunnel information: http://www.mbone.com.

[5] McCanne, S., Jacobson, V. Vetterli, M., Receiver-driven Layered Multicast ACM SIGCOMM, Aug. 1996, pp. 117-130.

[6] Stevens, W. Richard, TCP/IP illustrated, vol 1, Addison-Wesley,1994

[7] J.Bolot, T. Turletti, A rate control mechanism for packet video in the Internet, Proc. IEEE INFOCOM '94, June 1994, Toronto, pp. 1216-1223.

[8] C.K. Wong, S.S. Lam, Digital Signatures for Flows and Multicasts, IEEE/ACM Transaction on Networking, Vol. 7, No.4 August 1999.pp.502-513

[9] J.S. Boreczky, L.A. Rowe, Comparison of video shot boundary detection techniques. Storage and Retrieval for Image and Video Databases (SPIE) 1996: 170-179

[10]Marc H. Willebeek-LeMair and Zon-Yin Shae, Robust H.263 Video Coding for Transmission over the Internet, INFOCOM '98, Page(s): 225 -232 vol.1

[11]R.I. Chang, M.C. Chen, J.M. Ho and M.T. Ko, And Effect and Efficient Traffic-Smoothing Schema for Delivery of on-line VBR Video Streams, IEEE INFOCOM, 1999.

[12]Jaon L. Mitchell, William B.Pennebaker, Chad E. Fogg and Didier J. LeGall, MPEG Video Compression Standard, International Thomas Publishing.

[13]Bernard J.T. Jones, Low-Cost Outdoor Video Motion and Non-Motion Detection, Security Technology, 1995. Proceedings. IEEE 29th Annual 1995 International Carnahan Conference on , 1995 , Page(s): 376 -381

[14]Allen V. Oppenheim, Discrete-Time Signal Processing, Prentice-Hall Inc.,1989.