An Images Monitoring System based on Multi-Format Streaming Grid Architecture

Yi-Haur Shiau, Sun-In Lin, Shi-Wei Lo, Hsiu-Mei Chou, Yi-Hsuan Chen

***Abstract*—**This paper proposes a novel multi-format stream grid architecture for real-time image monitoring system. The system, based on a three-tier architecture, includes stream receiving unit, stream processor unit, and presentation unit. It is a distributed computing and a loose coupling architecture. The benefit is the amount of required servers can be adjusted depending on the loading of the image monitoring system. The stream receive unit supports multi capture source devices and multi-format stream compress encoder. Stream processor unit includes three modules; they are stream clipping module, image processing module and image management module. Presentation unit can display image data on several different platforms. We verified the proposed grid architecture with an actual test of image monitoring. We used a fast image matching method with the adjustable parameters for different monitoring situations. Background subtraction method is also implemented in the system. Experimental results showed that the proposed architecture is robust, adaptive, and powerful in the image monitoring system.

***Keywords*—**Motion detection, grid architecture, image monitoring system, and background subtraction.

I. Introduction

NOWADAYS, the network applications tied up with our life, such as mobile phone, PDA, image monitoring, VoIP, distance learning, etc. The network devices and technologies are developing continuously. At the same time, the image capture devices become more available today, and the image monitoring systems are getting more and more popular.

However, the traditional image monitoring systems transfer multi-media stream by Cable, and the systems must be bound with specific hardware devices. Most of the systems also have video record functions with a private video format that only can be played with their private players. Their monitoring results are stored on the video tapes. It is troublesome and hard to retrieve by the video tapes especially when the users are interested in specific events. Since the raised network bandwidth and the advanced network technologies, transferring multi-media stream by network becomes popular. Although there are some image monitoring systems which can transfer monitoring frames by network, and users can browse the frames through network. But it is still not enough for the monitoring purpose. And some other ones further support to detect the variation of the frames, and store these variations of dynamic frames into the database server for users to browse anytime. But the dynamic frames occupy a large amount of data storage.

In order to solve the problems the above-mentioned, a multi-format stream grid architecture for image monitoring system is proposed. It is a distributed computing, loose coupling and three-tire architecture, including stream receiving unit, stream processor unit, and presentation unit. Stream receiving unit supports multi capture source devices, such as HDV, DV, DC, WebCam, TV Card, Capture Card, etc., and also supports multi stream compress encoder formats, such as MPEG-1, MPEG-2 and MPEG-4. The benefit is it doesn't have to bind the specific hardware devices any more. Then, the stream clipping module of the stream processor unit clips the stream into static frame slices, and two consecutive frame slices are selected to proceed image matching. These frame slices are stored in the image database if the unusual variation situation is detected. The required space of data storage to store the static frames is much less than to store the dynamic frames. And for the presentation unit, there are three presentation platforms are developed, including standalone, web-based, and TDW ([1], [2]), a versatile, large, and high-resolution display system that was constructed by National Center for High-Performance Computing (NCHC for short), Taiwan, are developed. User can remote login with his account and password to browse the frames of the monitoring sites they have authorization by the web-based presentation platform.

This paper is organized as follows. In Section 2, we describe the proposed grid architecture. Section 3 describes the method of the image monitoring. Experiments with real data and results are presented in Section 4, followed by the conclusions in Section 5.

II. Multi-Format Stream Grid Architecture

The multi-format stream grid architecture we proposed is a three-tier architecture, which includes stream receiving unit, stream processor unit and presentation unit. Fig. 1 shows the multi-format stream grid architecture blocks and stream

Yi-Haur Shiau is with the National Center for High-performance Computing, Hsinchu, Taiwan (phone: 8863-5776085; e-mail: ihow@ nchc.org.tw).
Sun-In Lin is with the National Center for High-performance Computing, Hsinchu, Taiwan (phone: 8863-5776085; e-mail: lsi@ nchc.org.tw).
Shi-Wei Lo is with the National Center for High-performance Computing, Hsinchu, Taiwan (phone: 8863-5776085; e-mail: swl@ nchc.org.tw).
Hsiu-Mei Chou is with the National Center for High-performance Computing, Hsinchu, Taiwan (phone: 8863-5776085; e-mail: hmchou@ nchc.org.tw).
Yi-Hsuan Chen is with the National Center for High-performance Computing, Hsinchu, Taiwan (phone: 8863-5776085; e-mail: best1013@ nchc.org.tw).

pipeline. It is a distributed computing and loose coupling architecture. The benefit is the required numbers of server are depending on the system loading. When the image monitoring system monitors multiple sites at the same time, and loading is heavy, these three units can be distributed to different servers. On the contrary, they can be integrated into one server to save the hardware cost when the system loading is lighter. Based on the grid architecture, these servers can be any nodes on Internet, and it effectively enhances the system scalability. The functions of these three units are briefly introduced as follows.

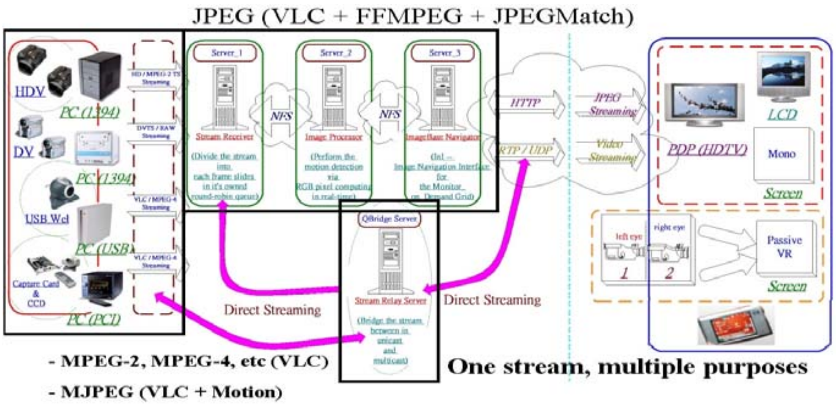

Fig. 1 Illustration of multi-format stream grid architecture blocks and stream pipeline

A. Stream Receiving Unit

The stream receiving unit can receive multi-format stream from the local monitoring sites or remote monitoring sites. Local monitoring sites transfer stream by cables, and remote monitoring sites transfer stream by networks. It supports multi capture source devices, such as HDV, DV, DC, WebCam, TV Card, Capture Card, etc. It also supports multi stream compress encoder formats, such as MPEG-1, MPEG-2 and MPEG-4. When the monitoring site needs high-resolution images to display the details, HDV or DV is selected as the capture source device. On the contrary, capture source devices with lower cost, such as WebCam or TV Card, are selected when high-resolution images are not required or the budget is limited.

Table 1 shows the relationship between the capture source devices and received stream formats. When the capture source device is HDV, the received stream format is HD/MPEG-2 TS. When the capture source device is DV, the received stream format is DVTS/RAW. The received stream format is VLC/MPEG-4 when the capture source device is WebCam, TV Card or Capture Card. In order to process different capture source devices and stream formats, the vlc ([3], [4]) and ffmpeg ([5], [6], [7]) functions are modified to automatically recognize capture source devices and stream formats.

TABLE I
RELATIONSHIP BETWEEN THE CAPTURE SOURCE DEVICES AND STREAM FORMATS

Capture source device	Interface	Stream format
HDV	1394	HD / MPEG-2 TS
DV	1394	DVTS / RAW DV
WebCam	USB	VLC / MPEG-4
Cable	Capture Card	VLC / MPEG-4

Because the most of traditional monitoring system are limited to bind with some specific capture source devices. The stream receiving unit in our system bring the benefit of that users can select suitable capture source devices depending on the image-resolution they need or the budget they have.

B. Stream Processor Unit

The stream processor unit includes three modules, stream clipping module, image processing module, and image management module. The functions of these three modules are briefly introduced as follows.

Stream clipping module. Clipping the stream data into consecutive frame slices is convenient to implement some image processing methods or some special effects on these frame slices. The stream clipping module is used to clip the stream received real-time from the stream receiving unit. These frame slices are put in the respective round robin queues for different monitoring sites. The stream clipping module modifies the vlc and ffmpeg functions to automatically recognize the stream format. Users can define the clipping frame numbers per second depending on the different applications.

Image processing module. The image processing module supports users to implement some kinds of image processing methods, such as image monitoring, event detection, image segmentation, object tracking and pattern recognition. In this study, we focus on image monitoring method in our system. Our system supports to monitor multiple monitoring sites at the same time, and the image monitoring method is parallel processed to the frame slices of the respective round robin queues of the different monitoring sites. Two consecutive frame slices are selected to process image matching, and to be detected whether unusual variation is occurred. In order to fulfill the higher resolution monitoring images, and the different monitoring sites with variable environments and situations, such as object movement or light variation, a fast image matching method with two adjustable principal parameters, RBG_Mean_Value_Difference_Range (RMVDR for short) and ScanLine_Match_Rate (SMR for short) is developed. The values of these two parameters can be real-time and on-line adjusted without interrupting the monitoring system, and it is suitable for different monitoring situations.

The image processing module in our architecture has some problems. The first problem is data lost due to the unstable stream transfer networks. The second problem is core dump happened when the data is read and written by the stream clipping module at the same time. The third problem is the matching false when the two consecutive matching frame slices are the first and the last files in the round robin queue. The last problem is the memory overflow due to ongoing monitoring all the time and the memory does not be released fully. The image processing module applies some methods to resolve these problems. For the first problem, our system implements some condition checks, such as the data format and size, to ensure the data are correct. If the data are incorrect, our system will enter to standby mode until the correct data is detected. For the second problem, to check the readability of data is implemented to ensure the data is readable only after the stream clipping module finished writing it. Otherwise, our system enters to standby mode until the readable data is detected. The third problem has happened due to our distributed loose coupling architecture. The stream clipping module and the image processing module are distributed on different servers depending on the different performance, and asynchronous situations may happen between these two modules. It causes the time to write the latter frame slice is earlier than the former frame slice. So, the time check function is implemented to ensure the time to write the latter frame slice is later than the former frame slice. About the memory overflow problem, our system is divided into two threads after loading the frame slices in the round robin queue. One thread is used to real-time load the latest frame slices, and replaces the old frame slices that have been processed with image matching in the background. It also implements effective memory management functions, including memory fully release and reallocation to avoid the memory overflow. The other thread is dedicated to do the image matching process.

Image management module. After finishing the image matching process of the two consecutive frame slices, and the unusual variation situation is detected. Our system automatically stores these two frame slices and relative log information. Depending on the different monitoring sites, these data are stored in the respectively management directories in the image database. By passing the image management server, users can query, browse, analyze, and manage these frame slices.

C. Presentation Unit

The presentation unit of our system includes three different presentation platforms, those are web-based, standalone, and TDW. Users can display the monitoring frame on their-owned PC with standalone platform, or display on TDW to monitor multiple sites at the same time. Moreover, a web-based presentation platform is developed for users to conveniently browse the monitor sites anytime and anywhere. The web-based presentation platform supports the security control. Users must use their-owned accounts and passwords to login, and they only can browse the monitor sites which they have authorization. Fig. 2 shows the web-based presentation platform. The left window shows the different monitoring options, such as monitoring sites and time etc. Users can browse the different historical monitoring image by selecting different monitoring sites or time. Users also can click the “live” button to browse the real-time monitoring images right now. The selected image is displayed in the right window. Users can click “DiffView” button to display these two unusual variation images at the same monitoring time. Fig. 3 shows the two unusual variation images. The left window shows the former image, and the right window shows the latter image. The relative log information is shown on the bottom window.

Fig. 2 Illustration of the web-based presentation platform

Fig. 3 Illustration of the web-based presentation platform – unusual variation images

D. Distributed Stream Compression

Two distributed video stream compression concepts are implemented to advance the streaming efficiency. One is the client-server based distributed compression, the other is server-side based distributed compression.

The client-server based distributed compression concept includes pre-compression in the client site and post-compression in the server site. Stream receiving unit receives video streaming from the client interface and implements pre-compression to reduce the network flow. The post-compress transfers the video streaming to multi-formats, such as MPEG-4, WMV, FLV and MJPEG.

The server-side based distributed compression concept is implemented to dynamically assign the video stream to the suitable servers depending on the loading of the servers. It compresses the video stream to a variety of bit-rates for different network bandwidth requirements. A real-time high resolution streaming system is implemented by using these two distributed stream compression concepts.

III. Image Monitoring Methods

One benefit of our image monitoring system is processing multi-resolution images and monitoring multiple sites at the same time. In order to carry out real-time image monitoring multiple sites with the multi-resolution images, a fast image matching method is implemented. In this method, two adjustable parameters, RMVDR and SMR, are used to be thresholds for proceeding image matching, and the value of these two parameters could be real-time and on-line adjusted to fulfilling different applications, such as objects moving or light variation. RMVDR, the RGB mean value of every scan-line of the frame slice, is used to match every scan-line of the opposite position of the two consecutive frame slices. If the value exceeds the defined threshold, it is identified these two scan-lines are different. Then, compare the numbers of the different scan-lines with the value of SMR. If the number value exceeds the SMR threshold, these two frame slices have unusual variation. These two frame slices and relative log information, such as the monitoring site, variation time, and variation rate, etc., are automatically stored in the image database. Due to image processing is a memory hungry application, system can easily run into crash without a proper memory management mechanism. We created two process threads: one dedicated to I/O handling and memory management, and the other for image matching process.

Another benefit is users can replace image processing module with their own image processing methods for other applications. Motion detection is an essential component of the video surveillance systems, and background subtraction is a popular approach for discriminating moving objects from the background ([8], [9]). In our implementation a real-time algorithm for background subtraction technology is adopted [10]. A reference background model is created by a set of sequence images and real-time updates it. Then, the current image is used to subtract the reference background model and detects the foreground object. Fig. 4 shows the experimental results of the background subtraction.

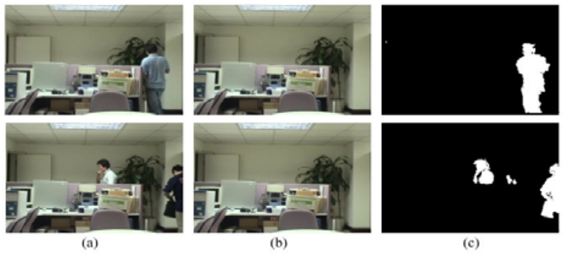

Fig. 4 Experimental results of the background subtraction: (a) the current video frames, (b) the reference background model, (c) the detection foreground objects

IV. Experimental Results

In order to real-time monitor multiple sites with multi-resolution images, over than 35 PCs are integrated into our grid architecture, and distributed to 5 kinds of servers. Fig. 5 shows PC cluster report using Ganglia monitoring system [11]. The first server is stream receiving server. It is used to receive multi-format stream from local or remote monitoring sites. The second server is stream-clipping server. It is dedicated to real-time clip stream into frame slices and stores them in the respective round robin queues depending on the different monitoring sites. The third server is qbridge server. Its function is to re-direct the stream from unicasting network to a multicasting network, and to exchange stream between unicasting and multicasting network. The fourth server is real-time unusual variation image matching server. It gets the frame slices from the respective round robin queues, and matching the frame slices in pairs. If the unusual variation situation is detected by our image matching method, these variation frame slices will be stored in the management directories of the respective monitoring sites. The last server is the web-based presentation server. It displays the images of the every monitoring site. Users must use their-owned accounts and passwords to login, and only can browse the monitoring sites they have authorization.

Fig. 5 PC cluster report by using ganglia monitoring system

In order to reveal the stream receiving unit supports multi-format stream, three different resolutions capture source devices, HDV, DV, and WebCam, are adopted in the demo cases. The first experiment uses HDV as the capture source device, and the image resolution is 1280x1080. The monitoring site is National Museum of Marine Biology and Aquarium (NMMBA for short), in Taiwan. We set up a HDV there, and the stream is real-time transferred to NCHC. It not only supports users to view the real-time frame, but also is used to do image monitoring. Fig. 6 shows the unusual variation when fishes move. Another monitoring site is the office in NCHC. Fig. 7 shows the unusual variation when someone opens the office door.

Fig. 6 Illustration of the unusual variation occurred in the NMMBA by using HDV

Fig. 7 Illustration of the unusual variation occurred in an office by using HDV

The second experiment uses DV as the capture source device, and the image resolution is 720x480. The monitoring site is a classroom in NCHC. Fig. 8 shows the unusual variation when someone walks through the classroom.

Fig. 8 Illustration of the unusual variation occurred of the pedestrian walks through the classroom by using DV

The last experiment uses WebCam as the capture source device, and the image resolution is 640x480. The monitoring site is an outdoor environment. Figure 9 shows the unusual variation when a trunk drives into the surveillance site.

Fig. 9 Illustration of the unusual variation of a trunk drives into the surveillance site using WebCam

V. Conclusion

This paper proposes a multi-format stream grid architecture for real-time image monitoring system. It is based on a three-tire architecture, including stream receiving unit, stream processor unit, and presentation unit. The benefit is the required numbers of servers are depending on the system loading. These three units are distributed to different servers when the loading is heavy. On the contrary, they can be integrated into one server to save the hardware costs.

In order to fulfill the image monitoring system with high-resolution images and based on distributed architecture, we use a fast image matching method with adjustable parameters for different monitoring situations of different sites. When the unusual variation is detected, the static images and related log information are automatically stored in the image database, and these images are displayed on web-based presentation platform. The benefits are efficiently processing high-resolution images and effectively save the data storage space. Some other image processing methods, such as background subtraction is also implemented in our system. In the future, more advanced image processing methods will be implemented to detect more events, such as face recognition and object tracking.

References

[1] S. I. Lin, F. P. Lin, C. L. Chang, S. W. Lo, P. Mai, P. W. Chen, and Y. H. Shiau, "Development of grid-based tiled display wall for networked visualization," Cellular Neural Networks and Their Applications, 2005 9th International Workshop on, 2005.

[2] TDW: http://tdw.nchc.org.tw.

[3] H. Nguyen, P. Duhamel, J. Brouet, and D. Rouffet, "Robust vlc sequence decoding exploiting additional video stream properties with reduced complexity," IEEE International Conference on Multimedia and Expo (ICME), 2004.

[4] VLC: http://www.videolan.org/vlc.

[5] T. Nakajima, "Realtime feedback and on-demand playback system for teaching skill improvement," Computers and Advanced Technology in Education, 2005.

[6] C. Traiperm, and S. Kittitomkun, "High-performance MPEG-4 multipoint conference unit," Networks and Communication System, 2005.

[7] FFMPEG: http://ffmpeg.mplayerhq.hu.

[8] S. Y. Chien, Y. M. Huang, B. Y. Hsieh, S. Y. Ma, and L. G. Chen, "Fast video segmentation algorithm with shadow, cancellation global motion compensation and adaptive threshold technique," IEEE Trans Multimedia, vol. 6, no. 5, pp. 732-748, Oct. 2004.

[9] D. S. Lee, J. J. Hull, and B. Erol, "A Bayesian framework for gaussian mixture background modeling," IEEE Proc. ICIP, vol.3, pp. 973-976, 2003.

[10] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, "Real-time foreground-background segmentation using codebook model real-time imaging," vol. 11, issue 3, pp. 167-256, Jun. 2005.

[11] Ganglia: http://ganglia.info.