

Method of recording video streaming in a network video monitoring system

5 TECHNICAL FIELD

[001] This invention is concerned with video monitoring system management, including the transmission, recording and storage of video footages. It is particularly concerned with video streaming from remote IP-cameras across
10 networks and the recording of the streams for video management and footage retrieval such as for monitoring and surveillance purposes.¹

BACKGROUND ART

15

[002] In the conventional DVR-based video monitoring systems (VMS), video input from the closed circuit TV (CCTV) tuner or camera's video capture card is sent directly to the digital video recorder (DVR) where the video is encoded and processed. This may be contrasted with the later NVR-based VMS, video
20 is captured by internet protocol or IP-based cameras where the video is encoded and processed, thus making the IP-video ready for streaming across networks or internet to the network video recorder (NVR) where it may be viewed centrally or stored. An NVR may dispensed with dedicated video capture or encoding hardware; hence, it may be based within a computer
25 network purely as software, remotely accessing and managing multiple live IP-video channels from the IP cameras at a central location.

[003] An NVR may have additional functions such as compression of the IP-video to reduce storage space needed for the video file, tagging of the IP-video
30 with metadata, etc. As a VMS, it is common for a NVR to be configured to

¹ Important not to mention any novel or inventive aspects (in this case, the prerecording part) in the Technical Field and Background Art sections.

record all the video streams at all times, i.e. 24 hours throughout the week (24×7) or even throughout the month, and thus requiring high capacity storage or disk usage which is costly. To reduce disk capacity or storage usage, the VMS user may consider configuring it to record only on an *ad hoc* basis, e.g. as per user selection to start recording upon seeing an interesting event unfolding before one of the displays or as captured by one of cameras, or which recording is triggered automatically by a detector such as motion, sound, vibration or light sensors, or combination thereof.

[004] In current NVR systems, electing *ad-hoc* recording in the aforesaid user-select or sensor-trigger methods would, however, result in the user missing out to record footage of the period immediately preceding the event or incident which started or triggered the recording. This period immediately preceding the event may be critical in providing the background understanding or information critical to the user's grasp of the event as recorded.

[005] It would thus be desirable to have a VMS, particularly an NVR system, which could be configured to automatically back-track the recording to a period immediately preceding the event. In this way, the user may choose to playback and view the recording prior to the triggering event to obtain a better understanding or information leading to the event.

LIST OF ACCOMPANYING DRAWINGS

25

[006] The advantages and features of our invention may be better understood when described with reference to the following drawings as exemplary and non-limiting embodiments, in which:

FIGURE 1 shows the input/output or data flow around a recording server employed according to our invention;

FIGURE 2 illustrates a broad configuration of various components of a general embodiment according to our invention; and

FIGURE 3 exemplifies a configuration of an embodiment of our invention deployed across a local area network.

5 DETAILED DESCRIPTION OF EMBODIMENTS

[007] Our prerecording process may be embodied as a software application in a Recording Server as shown in FIGURE 1. The software may comprise of 2 modules, i.e. a Prerecording Module wherein a Buffer Recording for
 10 prerecording is provided, and an Actual Recording Module. Present specifications allow for multiple IP-cameras up to 50 streams or channels at 256 kbps each may be configured. The video streams from the IP-cameras may be sent to the Recording Server by Real-time Transport Protocol (RTP) or, alternatively, via a Proxy Server or a Video Management System Recast
 15 Server (VRCA).

[008] The plurality of video streams received at the Recording Server may then be provided with 3 recording functions, i.e. (i) buffer recording for all the incoming video streams, (ii) actual recording and (iii) concatenating (i.e.
 20 stitching or joining) the beginning of the actual recording to its buffered recording when the actual recording is started or triggered. When the user start the recording, the Actual Recording starts the real-time, actual recording while at the same time packages the Buffer Recording into prerecording and concatenates the buffered recording to the start of the actual recording into a
 25 single video stream or file that is viewable and/or retrieval.

[009] The Recording Server may preferably be in communication with a VMS Video-on-Demand (VVID) Server (not shown in FIG. 1, shown in FIG. 2, *q.v.*). This direct communication enables recordings made at the Recording Server to
 30 be uploaded to the VVID Server by file transfer protocol (FTP). Preferably still, the Recording Server may also provide separate or parallel communication with a VMS Management (VMGT) Server whereby the

uploading to the VVOD Server's success or failure status may be monitored and updated accordingly.

[010] A broad or extended configuration of our invention comprising more components or alternative embodiments is shown in FIGURE 2 wherein data flow or processing therein are labelled Communication Flows, or simply Flow 1 to 10, for ease of discussion and may not be in accordance with the numeral's sequential order. At Flow 1, the video feed from the remote IP-cameras may be streamed to a VMS Recast Server (VRCA) or proxy server in Real-time Streaming Protocol (RTSP). The VRCA/proxy server may preferably be linked to VMS client, which may be embodied as a desktop PC or a security room console of multiple displays comprising a Command and Control System (CACS) workstation, as shown in Flow 2a.

[011] A user or security personnel on duty in the CACS room may initiate actual recording upon seeing an interesting or suspicious event. User interfaces for input such as joystick, touchscreen and/or keyboard may enable the user to specify which video stream to initiate recording. Alternatively, the actual recording may automatically be triggered by alarm as set off by a detector such as motion, sound, vibration or broken light sensors (e.g. broken infrared (IR) or far IR (FIR) beams) or other sensors, or combination thereof. Whether it is user-initiated or sensor-triggered request for recording, upon receiving such request, the VRCA/proxy server may send a copy of the video stream that has been requested to the VMS client to be viewed in real-time by the user, e.g. on the CACS workstation monitor.

[012] Contemporaneously, a copy of the requested stream to be buffered will be sent to a Recording Server, as previously described with reference to FIG. 1, in Flow 2b via Real-time Transport Protocol (RTP). At the Recording Server, in Flow 3, all the video channels or streams received from the VRCA/proxy server are buffered. Upon request, the Recording Server creates a recording of the video stream as per the request.

[013] To save disc storage, the buffer created for each of the streams will be for a predetermined length of time (e.g. 10 minutes) and cyclical, i.e. overwritten by fresh video stream if no recording request is received. It should be noted that the length of time may be varied, e.g. 5, 15, 20 or even 30 minutes, to provide a better visual or background understanding to the user or viewer of an incident or event before it happened. The longer the buffering time, however, will see the increase in memory and/or storage requirement. In fact, if the buffering time is sufficiently short, depending on the number of streams, video resolution and compression efficiency; the buffering may be entirely cached to reduce the need for read/write to storage disc.

[014] Another function of the Recording Server, as indicated in Stage 6 is to upload the successfully completed recordings to be stored at the VVOD Server (Stage 8). The uploading process may be performed in the File Transfer Protocol (FTP) across networks. The VVOD Server may perform a hash check to verify the uploaded video file's integrity and reports back to Recording Server of the successful upload, or failure if the upload fails as in Stage 9. The Recording Server may delete the local copy of the video file upon being reported that the upload was successful.

[015] The Recording Server may also update the VMS Management Server (VMGT) upon completion of recording uploads as shown in Stage 10. As shown in Stage 4, the VMS client, embodied as CACS workstation here, may preferably be provided with direct communication with the VMGT Server so that the user or security personnel manning the command & control room may signal to the VMGT Server to initiate recording of a particular video stream of interest. The VMGT Server may then issue a recording command to the Recording Server to record the specific video stream in Stage 5, whereby the Recording Server would record the actual recording and concatenates it with the prerecording from its buffer recording as well.

[016] It may be appreciated by a person skilled in the art, where any of the VMGT Server, Recording Server and VVOD Server are configured to communicate with each other remotely from each other across the network or via Internet, the communications mentioned in Stages 4, 5, 9 and 10 may be executed via conventional protocols, such as TCP/IP, which allows for ease of integration as well as having security features to be built-in within a broader communication platform such as Microsoft's Windows Communication Foundation's (WCF) endpoint-to-client connection within the .NET Framework.

10

[017] In FIGURE 3, a prototype of a Command and Control System (CACS) implementing our invention as configured for a local area network (LAN) at gigabit Ethernet capacity is shown. Such configuration should have useful implementation in advanced command and control systems such as the Singapore Civil Defence Force's (SCDF) Advanced C3 (Command, Control and Communications) Emergency System (ACES) in integrating various camera systems installed in different facilities or complexes.

15

20

[018] A hypothetical example may be illustrated in FIG. 3 of a remote secure zone in form of Jurong Island Fire Post having its own internal camera systems. Video streams from the cameras may be fed via Communication Flow (or simply Flow) 1 to a Video Recast Server (VRCA) running on the CACS local area network (LAN) of gigabit capacity. As shown in Flow 2a and 2b respectively, the VRCA Server will recast the video streams to the Recording Server as well as to the ACES client (which is shown as a VMS client). The Recording Server will, upon receiving a recording request from the VMS Server², start actual recording and stitch the corresponding prerecording with the actual recording. Upon finalizing the stitched recording, the Recording Server uploads the video file to the Video-on-Demand Server (VVOD) as

25

² This is as per Net2eAsia's original disclosure; but note that in FIG. 3, there is no VMS Server – there is only a VMS client (i.e. the CACS Workstation). Alternatively, as in the case of FIG. 2, the "VMS Server" could be a VMGT Server. So, please clarify.

indicated in Flow 3.

[019] While the methods as disclosed in our present invention have been tried and tested successfully with video streaming of digital or IP cameras, this is
5 mainly for ease of transmission of the video streams across networks as well as the internet in the standard format. The principle of prerecording from a cyclical buffer and stitching it to an actual recording should be applicable for different systems by suitable adaptation and/or modifications. For example, while we have successfully tested our video streaming using RTSP and RTP as
10 application layer protocols and UDP streaming at transport layer, thus MPEG-2 or MPEG-TS are supported, with suitable adaptation and modification, proprietary transport protocols such as RealNetworks' proprietary Real Data Transport (RDT) may also be supported once intellectual property rights (IPR) and technical specifications are cleared.

15 [020] Presently, the video codecs supported are the H.264/MPEG-4 AVC for the obvious reason of the free use of the open source H.264 encoder, our invention should be able to support other proprietary video codecs for the same IPR and technical reasons. Similarly, it may also be possible to adopt security
20 measures for the FTP upload, e.g. using secured FTPS (i.e. FTP with SSL/TLS) in our uploading process. These and other possible alternatives, modifications and adaptations to our aforesaid methods are to be considered as falling within the spirit and scope of our invention.

25 ***