Standardized and Cost-effective MPEG4 Video Streaming Server

J. MARTÍNEZ, B. MOLINA, C. PALAU, M. ESTEVE
Communication Department
Polytechnical University of Valencia
Camino de Vera S/N, Valencia. Zip Code: 46022
Spain

Abstract : - Current commercial video servers are implemented in a closed way. The video server is the most important device in a video surveillance system. All components in a video surveillance system depend on the video compression protocol and the encapsulation method used in the video server. The majority of commercial video servers are implemented with a dedicated DSP design. The hardware architecture and the video applications are usually closed in these devices. Furthermore, in commercial video servers, the video encoding protocols and encapsulation methods are also closed. This article shows a video server implemented with general-purpose hardware, open source software and standardized protocols. The proposed video server has been compared with commercial video servers and it has showed to be the best solution in terms of interoperativity, cost and scalability.

Key-Words: - MPEG4, Streaming, Video Server, COTS Hardware, Open Source, Embedded PC, SBC.

1. Introduction

Video encoding and video encapsulation are powerful functions. A video server must be a dedicated device that streams video uninterruptedly. The video servers are usually implemented with specialized hardware architectures. There are several general-purpose streaming solutions implemented via software, but many of them are not designed to work continuously.

Current personal computers are powerful enough to implement a video server function, but general-purpose operating systems and streaming software applications are not designed to implement dedicated and continuous real-time functions.

This article presents a video server implemented over a general purpose computer with a minimized free operating system and customized open source applications.

The SATRD[8] Group has implemented a video server with COTS hardware and open source software. This video server has been compared with commercial video servers resulting in the cheapest solution and obtaining similar results in terms of delay and robustness.

2. Motivation

The communication network in traditional video surveillance systems were based on point to point links and uncompressed transmissions.

Currently, video communication is implemented over packet oriented networks and video data is compressed before transmission [1][2].

IP networks are the most extended networks. Current switched IP networks are fast enough to support video transmissions. Switched IP networks are best-effort networks and do not guarantee all requirements of a real-time video surveillance system. In order to fulfil all requirements of these types of systems, video servers must implement complex tasks such as synchronization, timestamping, and ordering of packets.

The manufacturers of video server devices have adapted their hardware video servers to the new digital transmission networks and the latest video compression standards. Digital technologies and video standards evolve quickly and devices become quickly obsolete. The result is that hardware-based video servers are very expensive and become obsolete before their amortization.

Current COTS hardware with specialized software is able to implement the functions of a video server. The resulting system does scale economically and results in a cost effective video server solution that is able to upgrade its functions as new technologies and standards appear. One drawback of this combination is that specialized software can result expensive in a mid-size project.

The extension of multimedia applications in desktop systems and the growth of open source software have result in several free video software applications. These multimedia open-source applications implement the majority of the video server functionalities, but they are not aimed at implementing continuous real-time functions.

We are going to analyse the requirements of an industrial video server and to explain how to implement this device with COTS hardware and customized open-source software.
An analogue process can be followed with other hardware devices and the resulting solution (combination of COTS hardware and customized software) is often better than their equivalent hardware devices.

3. System Description

Commercial video servers have evolved towards DSP (Digital Signal Processor) architectures [3] [4]. DSP design is costly in hardware and software. In a DSP architecture, video compression and encapsulation must be implemented with DSP oriented software. This software is often the most expensive and closed part of the system.
Currently, some DSP systems are controlled by a simple operating system. This OS allows to reuse code between different projects. However, system calls, compilers and libraries are not standard and it is difficult to port an existent multimedia application from another operating system to the DSP system.
This section starts by describing the functional specification of a video server; after that the main requirements of a video surveillance system are introduced. Next subsection compares DSP architecture with an SBC (Single Board Computer) architecture. Finally the video server requirements are presented and studied.

3.1 Functional Architecture

The main functions of a video server are the following:

1. To capture video from an analog source.
2. To compress the video information.
3. To packetize the compressed video and to attach timestamps within the packets.
4. To transmit packets over the communication network.

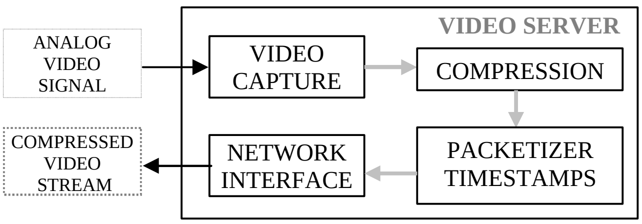

Fig.1 Simplified functional diagram of a video server.

- Video Capture: Video Surveillance systems usually require high-quality video and continuous motion (high framerate).
- Video Compression: There are several video compression standards [1] [5]. Currently, the most commonly used standards in video compression are MPEG4 [6] and AVL (MPEG4 part 10) [7].
- Packetization: IP networks are best-effort ones. These types of networks do not guarantee ordering, integrity and time synchronization. It is necessary to mark packets with timestamps and sequence numbers to ensure correct visualization. The most popular protocols of timestamping are RTP and MPEG System Transport Stream (MPEG TS).
- Network Module: Video communications can be unicast or multicast. In a video surveillance system it is typical to play and save a video flow simultaneously. In these cases it is recommended to implement multicast communications.

More advanced video servers implement extra functions such as voice communications and remote controlling of moving cameras. These functions will not be discussed in this article due to its extension.

3.2 Hardware architecture.

DSP systems are recommended for video and audio processing. They are based on RISC processors and use a dedicated architecture to process information very fast with a periodic pattern.

It is frequent to design a completely new DSP architecture for each project (due to efficiency reasons). Figure 2 shows one possible design.

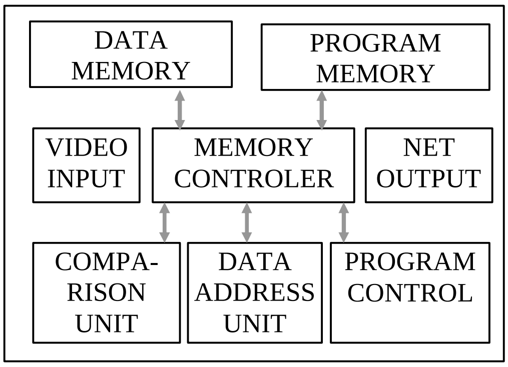

Fig.2 Example of a DSP architecture.

On the other hand, many SBCs have a conventional x86 architecture. Therefore, it is necessary a higher process capacity and memory resources to obtain the same efficiency that the one obtained with a DSP architecture.

Although SBCs are not designed to signal processing, the unitary cost of process-memory capacity is lower in an SBC architecture than in a DSP architecture. With an SBC it is possible to use a conventional operating system and available software with a considerably decreasing of the final cost.

The architecture of an SBC with Video Server capabilities is represented in the Figure 3.

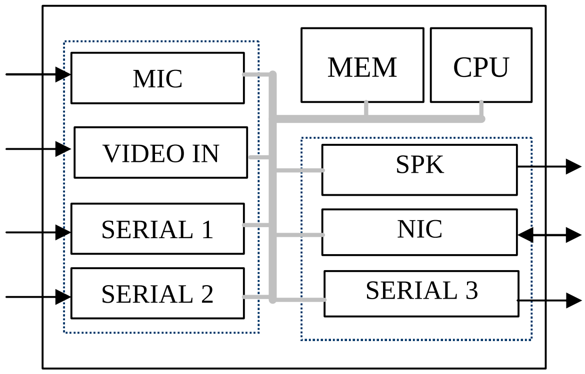

Fig.3. Example of SBC architecture.

3.3 Video Server Requirements

Though an SBC is not designed to implement video processing, the use of specialized software can accomplish typical video server requirements:

- High Video Quality (CIF at 25 fps or better).
- High Compression and reduced bandwidth utilization (less than 3Mbps each camera).
- Continuous operation mode.
- Short Delay (less than 300 ms).

With current conventional frame grabbers it is possible to capture up to 4CIF size at 30 fps.
High compression can be obtained with some free software video compression libraries (MPEG2, MPEG4 and AVL are implemented in available libraries).
The robustness (and continuous running) of the system can be obtained with the combination of a good hardware (industrial SBC) and an efficient Operating System like QNX or a customized version of Linux.
The requirement of a short delay is the most restrictive requirement. Software implementations are usually slower than hardware implementations. With a optimized operating system, fast encoding libraries, small video buffers and a fast switched network it is possible to obtain end-to-end delay lower than 300 ms.

4. Results

The SATRD[8] group from the Polytechnic University of Valencia has implemented a video server based on a standard low-range computer (similar to current SBCs) and open-source based software. The system fulfills requirements of industrial video servers.
The test system is an obsolete personal computer with the following characteristics:

- Processor: Intel Celeron 433 MHz
- Memory: 128 Mbytes
- Frame Grabber: Based on BT878 chipset.
- Video Input: Composite PAL.
- Operating System: Customized Linux Distribution (developed by SATRD).
- Video Quality:
  - Size: CIF
  - FPS: 25
  - Quantization Factor: 1
  - Delay: <200 ms
- Video Codification: Xvid (MPEG4)
- Video Encapsulation: MPEG TS
- Network Protocol: UDP

This system has been tested running continuously during three days obtaining the following graphical results shown in Figures 4 and 5.

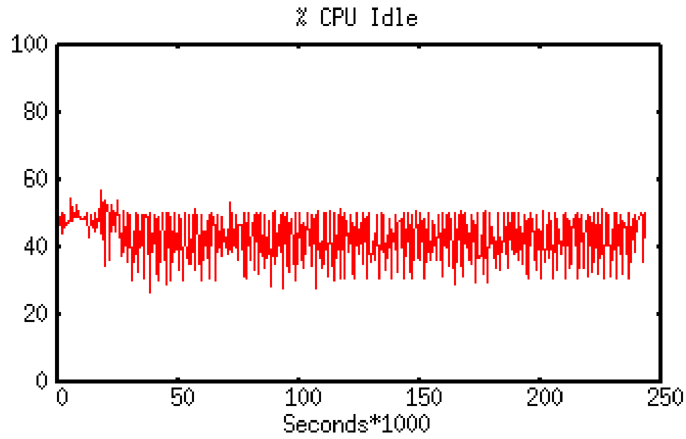

Fig.4 Processor Utilization

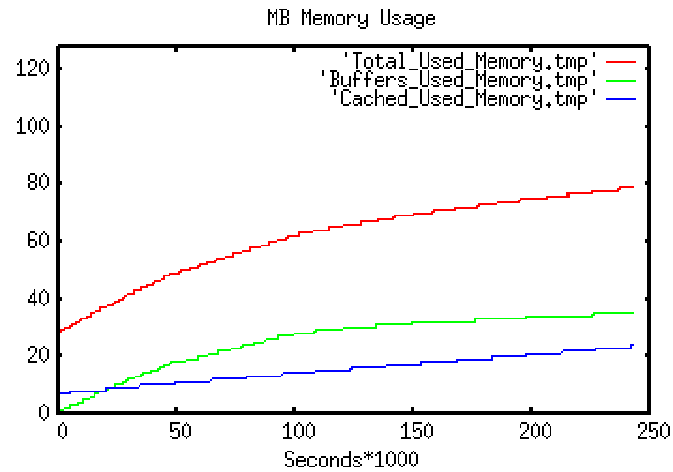

Fig.5 Memory Usage

The memory usage graphic (Fig. 5) shows memory divided into three data for each time: total memory, buffers memory and cached memory. It is usually that the operating system uses all the available memory in its buffers and cache. The main memory used by video application is:

$$\text{VIDEO MEMORY} \approx \text{TOTAL MEMORY} - \left(\text{BUFFERS MEMORY} + \text{CACHED MEMORY} \right)$$

This system has been compared with some DSP-based commercial video server. The comparison was based on subjective parameters obtained via an opinion poll realized to several operators of a wide video surveillance system. Our system obtained higher punctuation than all commercial video servers.

5. Conclusions

It is possible to implement a standardized video server using only COTS hardware and customised open source software.

Even using hardware with industrial features, final cost of the device is reduced considerably compared to commercial DSP based video servers.

Moreover, our system uses international standards in video compression and in the encapsulation protocol. Commercial Video Servers often implement closed video encoding protocols and the rest of devices in the surveillance system, such as the video player or the storage server are determined by this closed implementation.

The same principle followed with the video servers is extensible to other one-function devices. With COTS hardware, an optimised operating system, and customized free software it is possible to replace many of the expensive industrial dedicated devices.

References:

[1] T. Sikora, ¨MPEG digital video coding standards¨, IEEE Signal Processing Magazibe, Vol 14, Nº 5, Sept 1997, pp. 82-100.

[2] S. Rudkin et al, “Real-Time applications on the Internet”, BT Technology Journal, Vol 15 Nº 2, April 1997 , pp. 209-225.

[3] M. Budagavi et al, “MPEG-4 Video and Image Coding on Digital Signal Processors”, Journal of VLSI Signal Processing 23, 51-66, 1999, pp. 51-64.

[4] M. Bugadavi, et al, “Wireless MPEG’4 video on Texas Instruments DSP chips”, IEEE Proceedings Internationa Conference Acoust. Speech and Signal Process, March 1999.

[5] V. Bhaskaran & K. Konstantinides, “Image and Video Compression Standards: Algorithms and Architectures” Kluwer Academic Publishers, 1997.

[6] International Organization for Standarization, “MPEG-4 Overview”, ISO/IEC JTC1/SC29/WC11 – Coding of moving pictures and audio, March 2002.

[7] International Organization for Standarization, “Published international standard: Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding”. ISO/IEC 14496-10, 2004.

[8] SATRD: Group of Distributed Real Time Systems and Applications – Grupo de Sistemas y Aplicaciones de Tiempo Real Distribuidos – Polytechnic University of Valencia, http://www.dcom.upv.es/castellano/idi/grups_f.htm