

Software tools for the development of distributed intelligence surveillance networks

J. Viitanen and J. Jankkari

*VTT Technical Research Centre of Finland, Information and communication technologies,
P.O. Box 1300, 33101 Tampere, Finland*

Introduction, the scope of the work

Future surveillance systems will perform a number of automatic image and signal analysis operations. When the area of surveillance increases and large numbers of cameras and sensors are installed, the limits of the processing power in a centralized computing system are rapidly exhausted. Therefore future surveillance systems will most likely be based on distributed processing.

Other topics of demand for the distributed processing rise from the fact that digital cameras rapidly replace analog cameras, therefore the compression and encryption of data, and local adjustments already necessitate major local processing power, in addition to local intelligence. Efficient software tools are needed for the assessment and building of such distributed surveillance systems. This work reports some results from a European project SUBITO, where automated surveillance is being developed. The main objectives of the SUBITO project (Surveillance of Unattended Baggage and the Identification and Tracking of the Owner) are autonomous detection of unattended baggage and rapid identification, locating and tracking of the baggage owner. One of the work packages of the project was concentrating on supporting studies, and our task was to develop the hardware test platform and the software tools for the support when future distributed systems are being designed. Some aspects of these supporting studies are reported here. The actual identification, location and tracking are not addressed here.

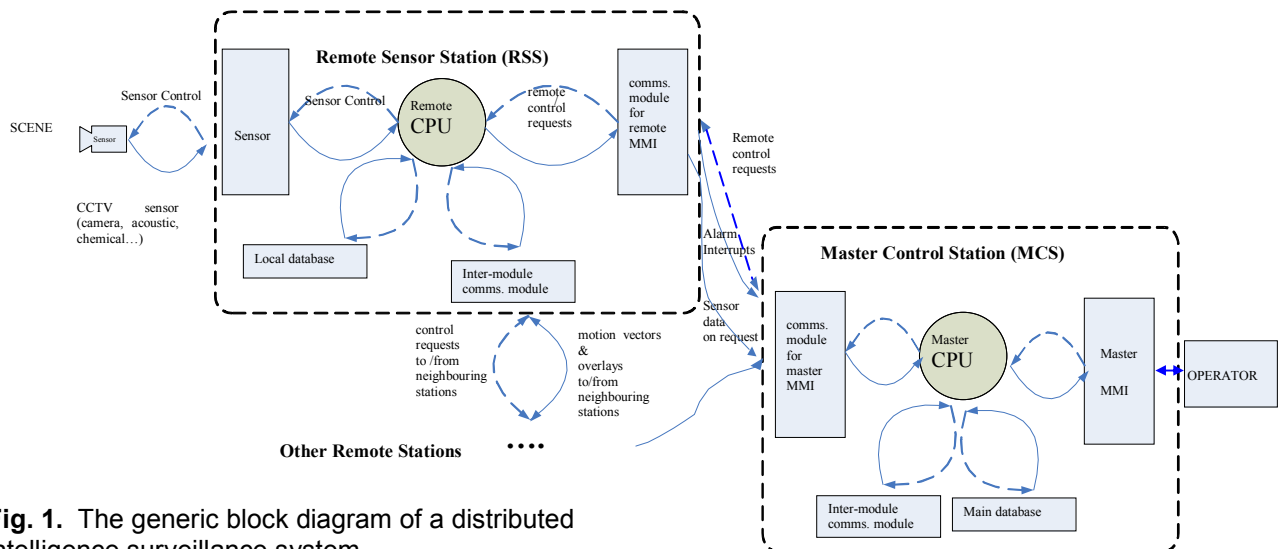


Fig. 1. The generic block diagram of a distributed intelligence surveillance system.

The methods

The methods of the work consisted of an analysis based on an experimental distributed surveillance system. In order to get a realistic view of the real bottlenecks and concerns for a large network, we built a system based on current commercial state-of-the-art components, especially with respect to wired and wireless communication facilities and high resolution

camera data rates, while the processing load was evaluated based on existing compression/decompression algorithms and the estimated automatic surveillance analysis task load distribution. For wireless communication, a few typical RF interference situations were activated. We assumed a typical situation where a central master surveillance station with a typical user interface was employed, with networked connections to the remote intelligent stations. Basic recognition was assumed to be done at the remote stations, but central command and data storage at the single master station. Figure 1. shows the block diagram of a generic test setup configuration, and Figure 2. the implementation with COTS components.

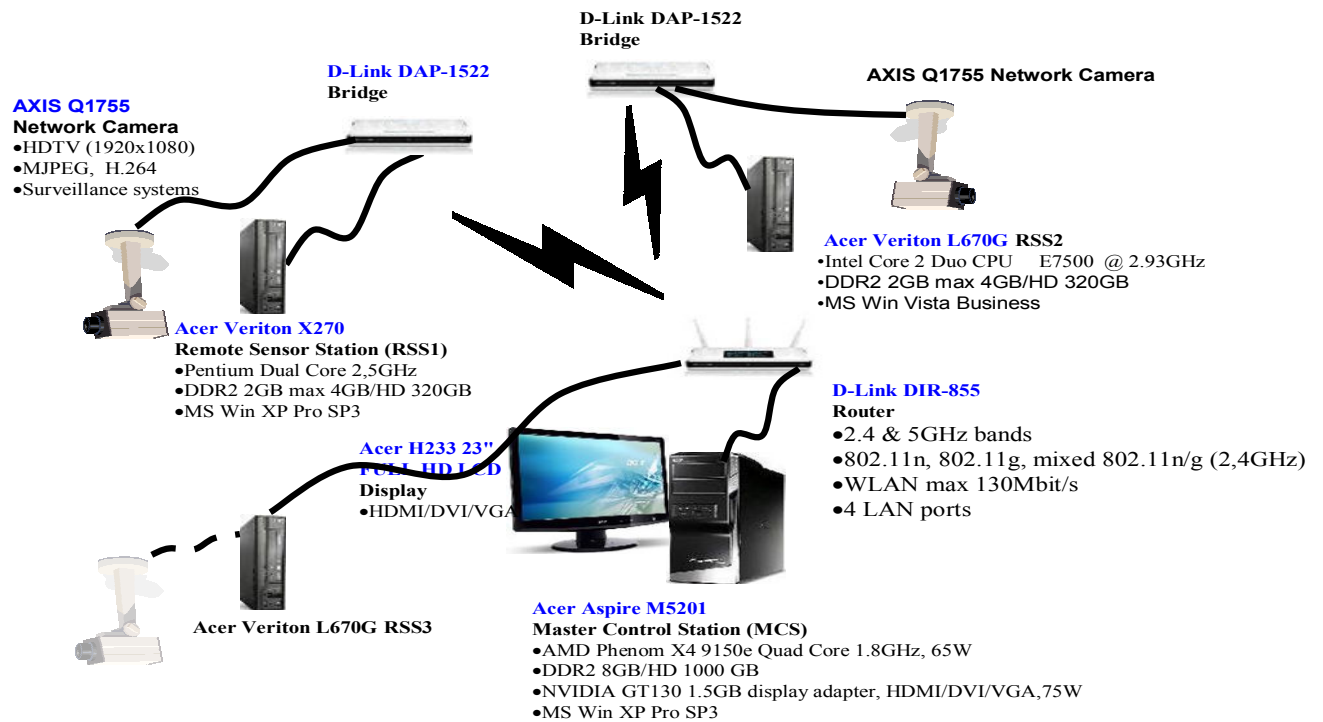


Fig. 2. The implemented test setup; two physical remote stations were implemented in hardware; larger networks can be implemented as virtual cameras. Both wired and wireless links were realized.

All the wired connections were typical 1Gbit/s Ethernet, save the 100 Mbit/s camera connections to the closest bridge. The performance and diagnostics tools used for the analysis of the software and connectivity performance consisted of the open source network analysis tool Wireshark, commercial Windows Vista performance tools, Task Manager CPU and Network load measurements, D-Link Router DIR-855 Diagnostic tools and debug time results. The most important functions of the system are:

- The identification or recognition of an alert situation (simulated in the test setup).
- Sending of the information forward in the system.
- Catching of the attention of the human operator for the necessary actions.

The primary actions initiated by these functions include:

- a) Decompression, simulated manipulation of the image frames, compression back again and writing locally to an alert file at the remote sensor station (RSS).
- b) Forwarding of the alert message with the alert video file to the master control station (MCS) for storing in the centralized alert history.
- c) Informing of the operator with the views to the alert video files and the alert history.

Figure 3. shows a graphical view of the software applications involved in these actions.

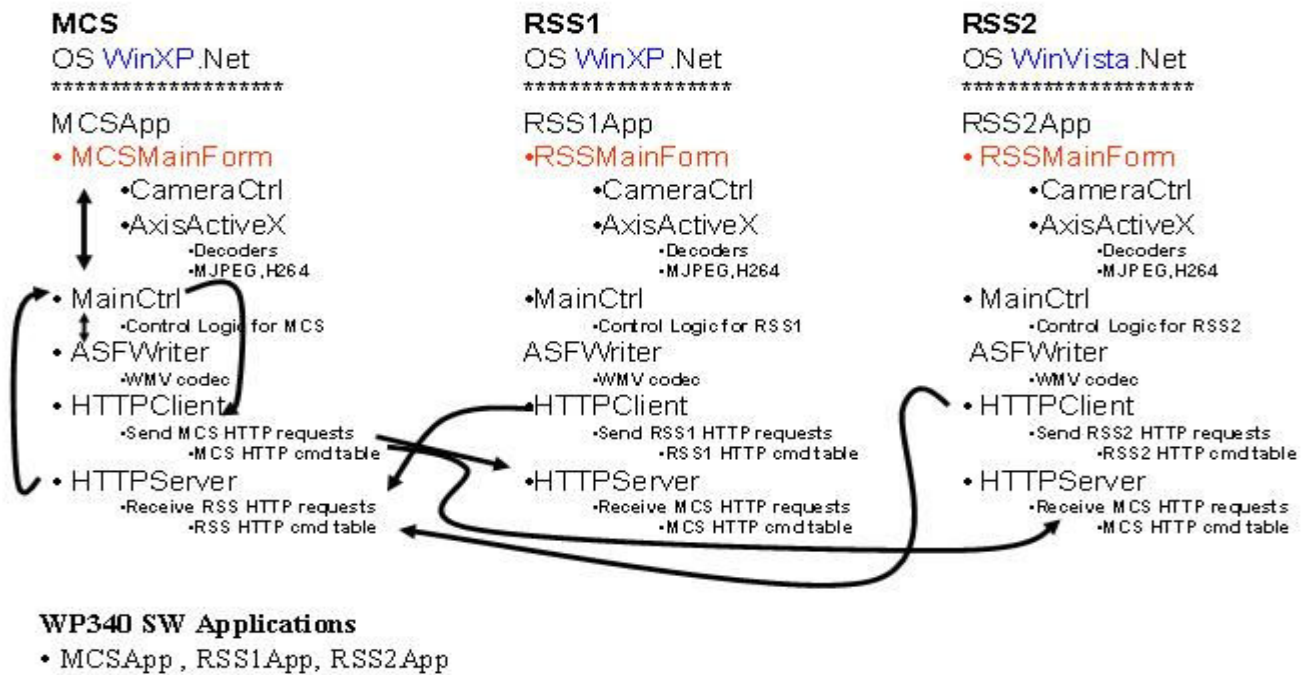


Fig. 3. The main software applications implemented in the test setup, and the main communication links between the master control station (MCS) and remote sensor station (RSS*) applications.

Results

The results of the analysis containing parameter tables relevant to the surveillance setups, such as camera resolution and corresponding data rates, available wireless and wired network bandwidths, CPU performance figures, and compression algorithm loads. The spare residual capacity left for the actual surveillance tasks was measured, and its sufficiency evaluated based on the estimated processing demand rising from the recognition tasks. The results concluded concrete requests for the optimal digital surveillance camera interface requirements, in a situation where the compression and encryption algorithms are embedded within an intelligent camera. Features were also tested for the design on the communication principles between the remote stations and the master control station. Additionally, implementation principles were proposed for the master control station user interface in cases where large numbers of cameras are used, and human performance is not sufficient for the actual recognition, but still service for rapid requests of specific attention is required by the security personnel. Due to the limited space of this abstract, in the following we have just a short digest of the main findings from the performance measurements.

1. Camera – RSS tests

The Axis Q1755 network camera was started to stream video data in two alternative compression formats, in MJPEG or in H264. The compression degree of all Camera – RSS tests was 30 in the 1...100 scale. The following operations were performed: retrieving the video, decompressing, manipulating, compressing back and writing to a file. The two operating systems tested were Windows XP Pro and Vista, and the compression and decompression routines were provided by the camera manufacturer. The average decompression time per frame included the actual decompression and the data move to a buffer, and the average compression time per frame included light data manipulation (a red alert text transmitted to the master station) plus the compression in the debug mode of MS

Visual 2008. For the highest MJPEG resolution, 1920 by 1080, the decompression process took 70ms per frame at 73% CPU load, while for the highest H264_1080i resolution (1440 by 1080), it took 95ms per frame at 96% CPU load. Compression times were less than half of those times for the same resolutions. The tests with real image analysis and tracking algorithms are still due, but the compression/decompression algorithms are quite representative for heavily loading tasks where every pixel needs to be processed, while for the typical image analysis tasks, the amount of data rapidly drops and the speed per frame correspondingly increases in higher level processing after initial feature extraction and segmentation.

2. RSS to MCS data transfer tests

For the data transfer tests between the remote and master control stations, the wireless results offered lower performance, compared to the high speed gigabit Ethernet wired network. The new network bridges that offer mixed 2.4GHz IEEE802.11g and 5GHz IEEE802.11n wireless links are advertising up to 600 Mbit/s radio link rates. In our short distance tests up to 50m distance, the highest net throughput, however, was 48 Mbit/s. Also, with wireless links it is natural that if there are more stations sharing the same channel, the speed drops correspondingly. Generally the throughput varied in the wireless tests between 12 - 34 Mbit/s. The wired file transfer tests showed 2-3 times better throughput over the wireless technique, i.e. transfers from RSS to MCS in the 1Gbit/s link, an average speed of 192Mbit/s. Remembering the improved compression of H.264 over MPEG2, the bandwidth requirement of HDTV transfer has dropped from about 15Mbit/s to about 8Mbit/s. So both the wireless and wired links with suitable buffering will be able for real time transmission of several channels; however, with just a few other wireless devices sharing the same frequency band, the wireless channel soon becomes overloaded.

Conclusions

Both the compressed and decompressed image formats are needed in camera surveillance systems: compressed formats for storage and transmission, and decompressed for image analysis. Doing compression/decompression by software using the modern complex compression standards is overloading typical present consumer PC processors. The situation is different in the centralized vs. the distributed processing cases. A centralized system using high resolution digital cameras typically receives the data compressed. For decompression which is a heavily loading operation, the cost of using special hardware for lower loading of the CPU is typically not very critical. However, for a distributed system, it is attractive to use low cost, but high performance PC-type processors, and additions of dedicated decompression modules to each of them increase costs considerably. Being close to the camera, the raw data would be readily available. Many digital surveillance cameras have the option to provide either the compressed or raw data, **but not simultaneously both**. "Smart" cameras have an on-board local CPU, but often it would be desirable to leave that for the routine camera controls and compression, and on the other hand, reserve the local external CPU for the image analysis tasks. That would require some additional bandwidth for the camera for simultaneous output of both the raw and compressed data.

From the wireless transmission tests, the conclusions are quite clear. Our applications are targeted for crowded areas where there are potentially many other WLAN users, and often for safety critical tasks. Therefore the deterioration seen in case of several WLAN users (or deliberate jamming by whatever transmitters) is intolerable. In principle, the use of high gain directional antennas could improve the situation for fixed longer distance links, however, also for such links high level of external RF noise is possible.