ENERGY CONSUMPTION AND IMAGE QUALITY IN WIRELESS VIDEO-SURVEILLANCE NETWORKS

Carla-Fabiana Chiasserini and Enrico Magli

CERCOM - Center for Multimedia Radio Communications
Dipartimento di Elettronica - Politecnico di Torino
Corso Duca degli Abruzzi, 24 - 10129 TORINO - ITALY
Ph.: +39-011-5644195 - Fax: +39-011-5644099
E-mail: {`chiasserini,magli`}`@polito.it`

Abstract— Wireless video-surveillance networks are gaining increasing popularity due to the number of applications they make possible to carry out. In this paper, we address the problem of designing a wireless video-surveillance network so as to optimize its performance. In particular, we investigate the possible trade-offs between energy consumption and image quality. We provide simulation results showing that video compression can be very beneficial in reducing data transmission costs, provided that the energy cost of video compression is low. Moreover, we discuss the impact of compression on image delay.

Keywords— wireless sensor networks; image and video compression; low-power algorithms; wireless network architectures;

I. INTRODUCTION

Wireless sensor networks are rapidly emerging as a framework to carry out distributed and pervasive applications such as environmental monitoring, domotics, marine biology, habitat studies, and video-surveillance, just to mention a few. Such networks typically consist of tens to hundreds of small-size and low-cost nodes; each node is equipped with a sensing device that collects information from the environment (e.g., temperature, vibrations, images), and transmits it through the network to a gateway node, where these data are analyzed and the desired information is extracted [1]. Amongst all possible applications of sensor networks, in this article we focus on wireless video-surveillance networks (WVN), that is, sensor networks in which each node is equipped with a camera that frames a video scene of the region of interest [2].

In a WVN, energy consumption is a critical issue [3], even more than in other sensor networks since video cameras collect a huge amount of data that must be transmitted over the wireless link. In principle, video compression can reduce the amount of data by a considerable factor. On the other hand, it is well-known that most video coders exhibit a very high computational burden. This is not a matter of concern in desktop multimedia applications, in which one can afford a 2 GHz processor to encode and decode video at 30 frames per second in real-time. While, the use of such a powerful encoder in a WVN application may be objectionable, in that it is possible that the energy saved by transmitting less data does not compensate for the energy required to compress the video data. The trade-off between communication and computation is therefore a crucial aspect that need to be investigated.

Another issue in WVNs is the delay incurred by the video stream due to the processing time and the latency in the network. Intuitively, one would think that compressed video should exhibit lower delay than uncompressed video. However, if the sensor processing unit is not sufficiently powerful, the delay can be dominated by the processing time. Keep in mind that, while several compression algorithms employ floating-point arithmetic, most low-power processors have a fixed-point architecture, which requires several clock cycles to emulate a floating-point operation.

Additionally, in the design of a WVN, one has to carefully account for the data quality issue. While video compression algorithms often have very powerful data reduction capabilities, they do introduce distortion. When the network gateway decodes the video stream and analyzes the data, the distortion introduced by compression can heavily bias the results, thus possibly leading to wrong interpretation. Note that, in case of still images, although current standards as JPEG (Joint Photographic Experts Group) and JPEG2000 can yield compression ratios in excess of 30:1 with good visual quality, it is generally believed [4] that a ratio in excess of 10:1 can significantly alter the results of automatic image analysis algorithms to be run on the decoded images.

Thus, several factors must be taken into account in the design of a WVN, and primarily energy consumption and image quality. These factors heavily interact with each other, thus making it difficult to find an optimal trade-off. The goal of this paper is to investigate possible solutions to this trade-off, by identifying suitable compression algorithms, assessing the communication and computation costs, and taking into account the quality of service requirements in a WVN.

II. WIRELESS SURVEILLANCE NETWORKS

In this section, first we describe the system scenario and report the system characteristics and parameters. Then, we focus on algorithms for video compression and evaluate the cost of

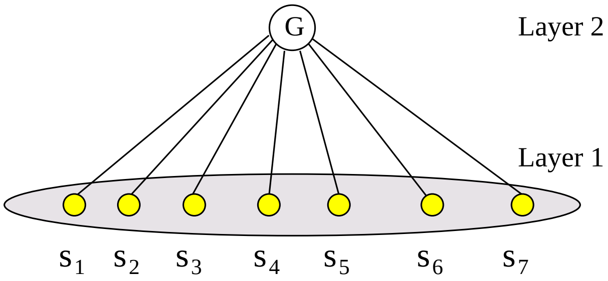

Fig. 1. Hierarchical network architecture.

such algorithms in terms of energy consumption.

A. SYSTEM SCENARIO

We consider a wireless surveillance network composed of a set of stationary, battery-powered, sensor nodes. For the sake of simplicity, we assume that the area under observation is an orthogonal surface with respect to the plane where sensors are located. All sensors are alike, each of which is equipped with a camera, a low-power microprocessor, and a radiofrequency (RF) circuitry that allows sensors to communicate over the wireless channel. Data observed by the sensors are collected at a node, denoted by G, which is either a central controller or a gateway to the fixed network.

Let p_t denote the output power necessary for transmitting one information bit and p_e denote the power consumption of the RF circuitry per bit, which is the same in transmit mode as in receive mode. Let C be the battery capacity, expressed as Ah, that is initially available at each sensor, and let P_{max} be the maximum power that can be drained from the battery. In our network scenario, all sensors have same processing capability and initial energy resources. The system characteristics and parameters values are presented in Table I.

We assume that the network architecture is as shown in Figure 1, where the highest layer of the architecture is represented by node G. Nodes at the lowest layer, indicated by $s_i, i = 1, \dots M$, are in charge of monitoring the area of interest and are positioned along a line in such a way that the target area is fully covered. Each of the M sensors observes a portion of the target area and forwards the video signal toward G. We assume that sensors camera produce grayscale QCIF (Quarter Common Intermediate Format) images (i.e., 144 lines of 176 pixels each), which are fragmented at the link layer into Protocol Data Units (PDUs) and transmitted over the wireless link. The radio channel is modeled as a Gilbert channel [5], with two states, *good* and *bad*, that represent the channel conditions during the transmission time of one PDU. The average data unit error probability is denoted by ϵ and is taken as a variable of the system, while the average burst of error over the channel is set to 5.

We consider an ARQ scheme implemented at the link layer, so that the sender transmits a PDU until either the PDU is correctly received or a maximum number of transmission attempts is reached. We denote the maximum number of transmissions per PDU by N_t. A PDU is discarded if it is not successfully transmitted within N_t attempts. In the following, we assume that a non-compressed image is lost if more than L PDUs are discarded, while a compressed image is lost as soon as one PDU is discarded. We denote the image loss probability by P_I. The effect of PDU losses on the uncompressed and compressed data is discussed in the next section.

B. VIDEO COMPRESSION TECHNIQUES

In the WVN application, an important issue is the selection of the coding framework that best matches the characteristics of the video-surveillance data and the energy constraints of the application. The most popular video coders defined by the ISO (International Standardization Organization) and ITU (International Telecommunication Union), i.e., MPEG-2, MPEG-4, H.261, H.263, and the novel H.26L, are usually referred to as hybrid motion compensated coders, since they perform motion estimation in order to capture temporal redundancy by following object motion within the scene. This approach is known to yield very good results but is also very time-consuming, especially in the motion estimation stage. On the other hand, it is worth noticing that, unlike typical multimedia video, surveillance video sequences are often characterized by low motion, in particular when no object is expected to move within the scene but in case of anomalies. Thus, such coders do not provide a favorable energy/performance trade-off in the WVN application.

In the context of surveillance of a low-motion scene, the following approach can be pursued. Each sensor can periodically encode and transmit a reference frame (e.g., one scene every 15 seconds) and then, between two consecutive reference frames, transmit only differences (if any) between the current acquired frame and the reference one. In this way, since the scene should be still unless an anomaly occurs, no energy consuming motion estimation is performed, whereas temporal redundancy is captured. Also, since scene changes are supposed to occur rarely, very few refresh data have to be transmitted allowing for a very high compression ratio.

In order to select a suitable encoder for the reference images, we have considered several existing image compression algorithms, including JPEG and JPEG2000, and we have evaluated the trade-off that they provide between energy consumption and compression ratio. We have employed a simulator of the ARM architecture, along with an accurate energy estimation tool for the StrongArm 1110 processor [6], in order to characterize the energy cost of each algorithm. We have found that, in general, algorithms that employ a floating-point computational engine, such as JPEG2000 and JPEG with the floating-point discrete cosine transform (DCT) kernel, exhibit a very high energy consumption. Thus, their actual usefulness for the WVN application is fairly questionable. On the other hand, when fixed-point kernels are used, as in JPEG with the integer kernel, the energy cost becomes as low as to provide a favorable energy-compression trade-off. As an example, encoding a grayscale QCIF image at 1 bit-per-pixel, employing JPEG with

TABLE I

SYSTEM CHARACTERISTICS AND PARAMETERS.

System Component	Performance
Microprocessor	Intel StrongArm 1110 @ 59 MHz
Battery	C=0.65, 2.7 Ah; P_{max}=20,50 dBm
JPEG	Compression Factor=8:1 Integer DCT kernel greyscale QCIF image size
Power consumption [7]	$p_t = 100\, pJ/\text{bit}/m^2$; $p_e = 50\, nJ/\text{bit}$
Uncompressed image size	25344 bytes
Reference image capture period	15 s
Link Layer PDU size	32 bytes
Transmission Bit Rate	200 Kbps
Maximum acceptable image delay	2 s

the integer kernel and using a SA-1110 processor at 59 MHz, requires 2.87 mJ and takes 89.8 ms. The same operation with the floating point kernel requires more than 22 mJ, which is a figure comparable with the transmission cost of the uncompressed image. As a consequence, in our simulation study, we have selected JPEG with integer kernel as compression algorithm. In general, we have found that few low-energy image and, especially, video compression algorithms have been proposed in the literature so far. In Fig. 2, the effect of PDU losses on the uncompressed and compressed data is shown. Fig. 2-a shows an uncompressed QCIF video-surveillance image representing a tunnel. Fig. 2-b reports the JPEG 8:1 compressed and decoded image when the integer kernel is used and restart markers at the end of each row of 8×8 block are inserted. In Fig. 2-c, the effect of losing 2 PDUs on the uncompressed image is shown; in this case, a PDU simply contains 32 consecutive image pixels. As can be seen, image quality considerably degrades after few PDUs have been lost, thus justifying the definition of P_I given in the previous section. Fig. 2-d reports the effect of one PDU loss on the JPEG compressed stream; upon detection of a corrupted packet, the decoder must wait for the next restart marker to resynchronize and restart decoding. It is clear that, in the compressed case, one single PDU loss heavily degrades image quality.

III. SIMULATION RESULTS

We derive the system performance in terms of image loss probability and average energy consumption per image. We consider the radio connection between one sensor node and the gateway node, G, and we compare the results achieved with uncompressed images to those obtained when the JPEG algorithm is used. The system parameters are set to the values reported in Table I.

Fig. 3 shows the image loss probability, P_I, as the maximum number of transmission attempts at the link layer varies and for different values of error probability over the wireless channel. The left plot presents the results in the case of uncompressed images, with $L = 2$ and 4; the right plot shows the results obtained by using video compression. We consider that, in order to provide a good image quality, P_I has to be below 10^{-2}. By fixing the desired P_I and looking at the plots in Fig. 3, we can obtain the required value of N_t as the error probability, ϵ, changes. Clearly, the image loss probability decreases with the increasing of N_t and the decreasing of ϵ; and, in the case of uncompressed images, a lower P_I can be obtained by using higher values of L.

Fig. 4 shows the average energy consumption per successfully transmitted image, as N_t and the error probability over the wireless channel change. Both the cases of uncompressed and compressed images are considered. In the case of compressed data, energy consumption includes the energy spent for communication as well as for compression. As shown in the plots, for low values of N_t, we can reduce energy consumption by increasing the number of transmission attempts. In fact, by increasing N_t, the probability that a PDU is discarded decreases, and, hance, the number of successfully transmitted images grows. For higher values of N_t, energy consumption does not further decrease by increasing N_t. This suggests that, once the desired value of P_I is achieved, there is not reason to increase N_t beyond the value that guarantees the minimum energy consumption per image. Finally, the plots in Fig. 4 show that, for all the considered values of L and ϵ, a lower energy consumption is obtained when images are compressed.

Results concerning the image delay were also derived (they are not shown here for the sake of brevity). Image delay is defined as the period from the time instant when the transmitter starts compressing an image to the time instant when the last PDU of the image is correctly received. Results showed that the network system always provides low image delay relatively to the inter-image period of 15 s and to a reasonable refresh period of 1-2 s, and that, for all values of L and ϵ, a lower delay is achieved when video compresion is used.

IV. CONCLUSIONS

We studied energy consumption and image quality in wireless video-surveillance networks, when retransmission of cor-

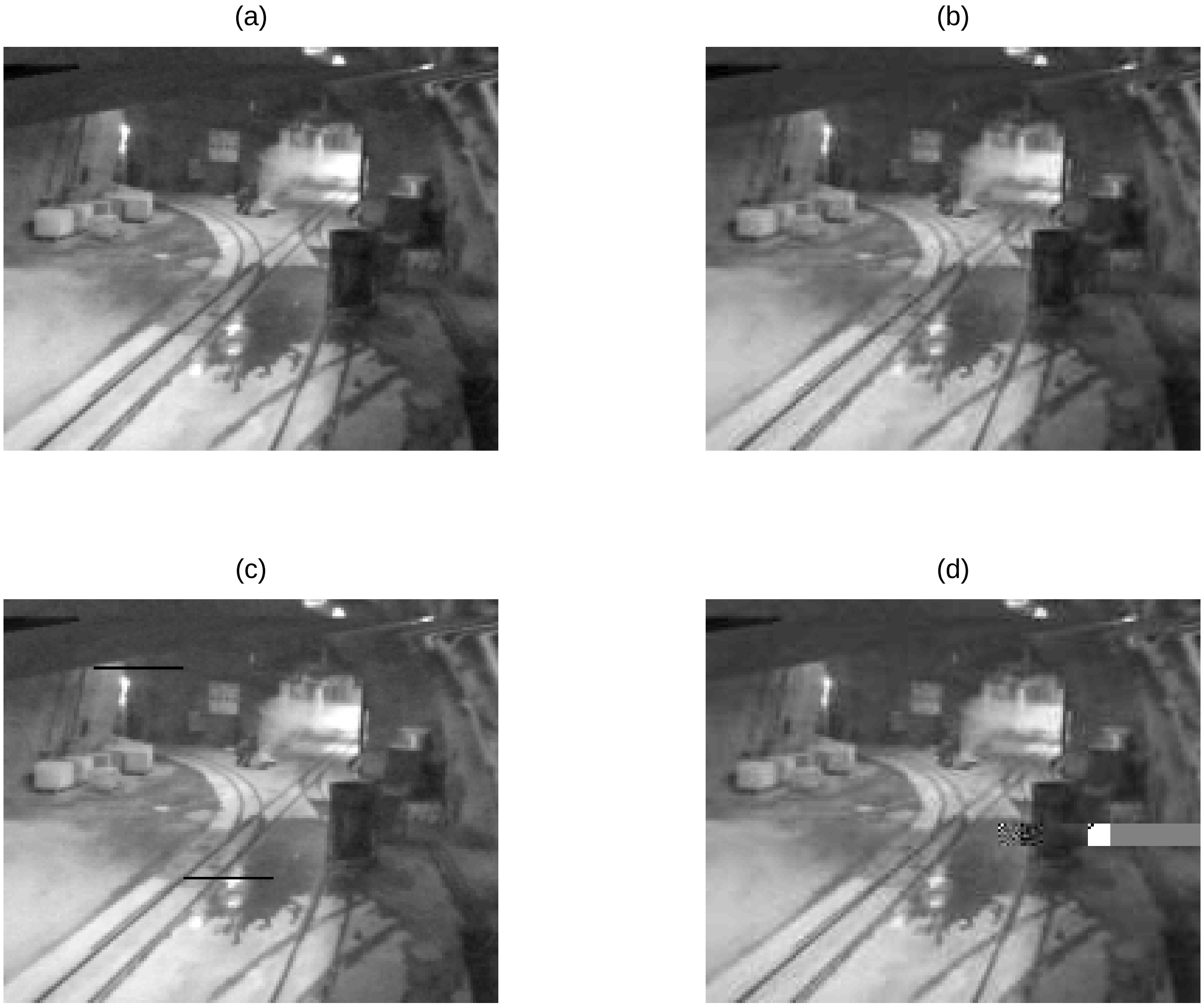

Fig. 2. Effect of JPEG compression and PDU losses. (a) Original image; (b) 8:1 compressed image; (c) received uncompressed image with 2 lost PDUs; (d) received compressed image with 1 lost PDU.

rupted packets is performed. Simulation results showed that video compression is indeed beneficial in reducing transmission costs, as well as image delay, provided that its energy cost is low, e.g., in the case of JPEG with integer DCT kernel. Moreover, the energy cost of retransmissions turns out to be sustainable. In particular, it is convenient to increase N_t so as to work in the flat part of the energy curves reported in Fig. 4, and as long as the desired value of the image loss probability is guaranteed.

REFERENCES

[1] K. Sohrabi, J. Gao, V. Ailawadhi, G.J. Pottie, “Protocols for self-organization of a wireless sensor network,” *IEEE Personal Communications*, Vol. 7, No. 5, Oct. 2000, pp. 16–27.

[2] R.T. Collins, A.J. Lipton, H. Fujiyoshi, T. Kanade, “Algorithms for Cooperative Multisensor Surveillance,” *Proceedings of the IEEE*, Vol. 89, No. 10, pp. 1456–77, Oct. 2001.

[3] J.L. da Silva Jr., J. Schamberger, M.J. Ammer, C. Guo, S. Li, R. Shah, T. Tuan, M. Sheets, J.M. Rabaey, B. Nikolic, A. Sangiovanni-Vincentelli, P. Wright, “Design methodology for PicoRadio networks”, *Proc. of DATE 2001*

[4] J.A. Saghri, A.G. Tescher, J.T. Reagan, “Practical transform coding of multispectral imagery”, *IEEE Signal Processing Magazine*, pp. 32-43, Jan. 1995

[5] M. Zorzi, R.R. Rao, L.B. Milstein, “ARQ error control for fading mobile radio channels,” *IEEE Transactions on Vehicular Technology*, Vol. 46, No. 2, May 1997, pp. 445–455.

[6] A. Sinha, A.P. Chandrakasan, “JouleTrack - A web based tool for software energy profiling”, *Proc. of Design Automation Conference*, pp. 220-225, 2001

[7] W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan, “Energy-Efficient Communication Protocol for Wireless Microsensor Networks,” *Proc. of Hawaiian International Conference on System Science*, Jan. 2000.

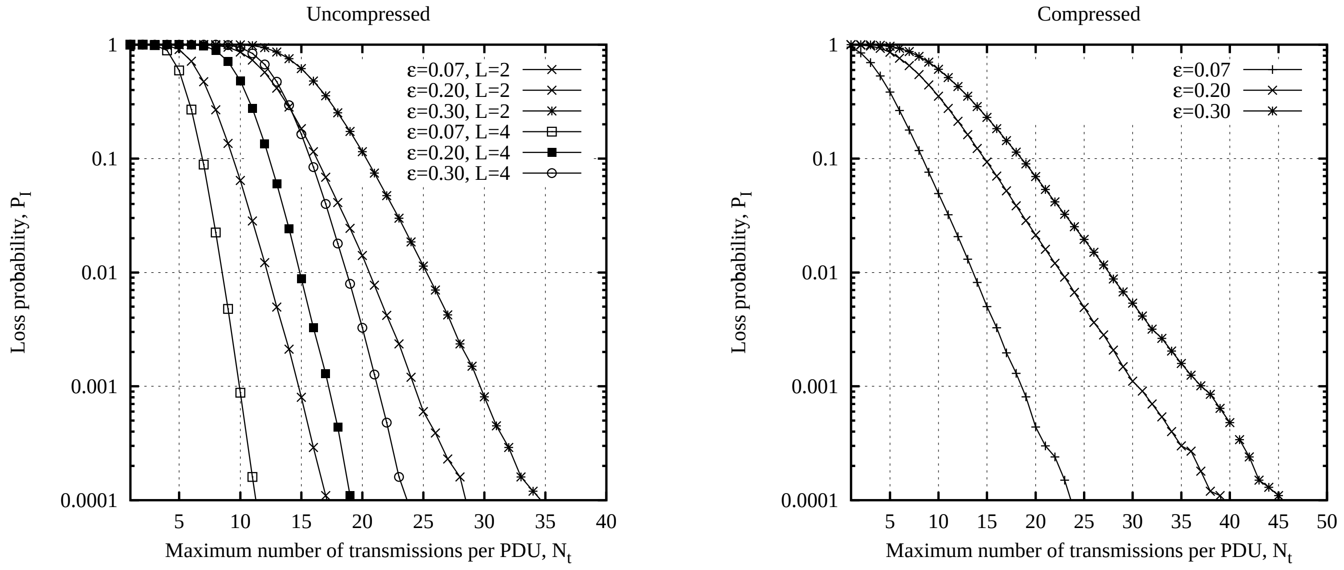

Fig. 3. Image loss probability versus maximum number of transmission attempts, as the the average error probability over the wireless channel varies. The case of uncompressed images with $L = 2, 4$ is presented in the left plot; the case of compressed images is shown in the right plot.

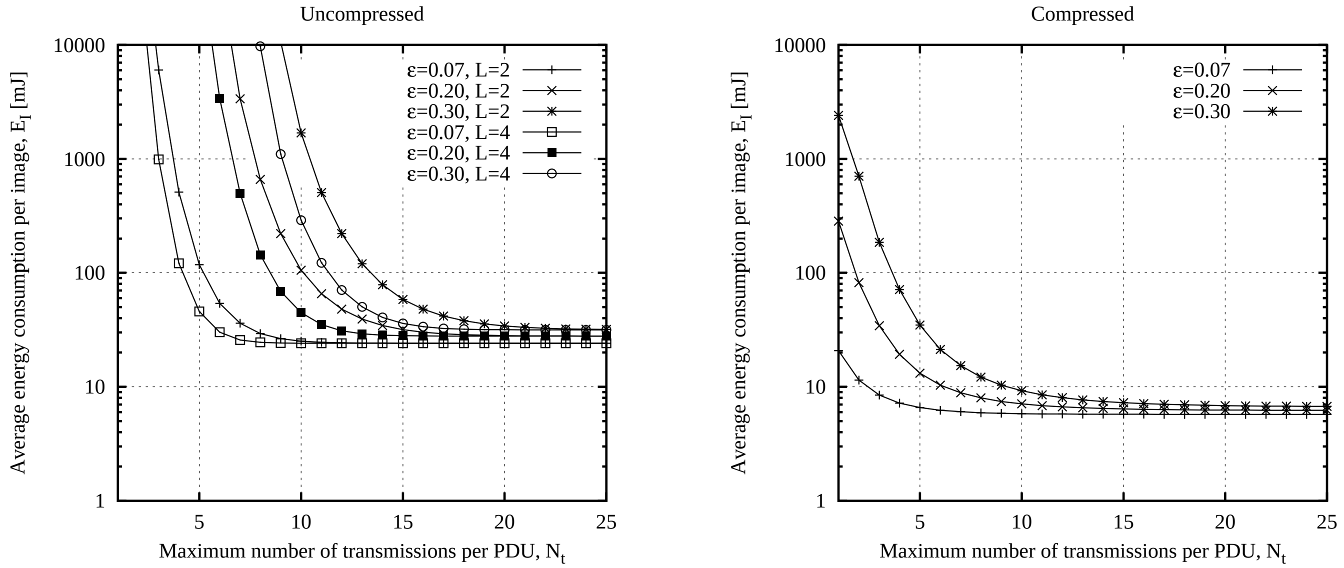

Fig. 4. Average energy consumption per correctly received image as the maximum number of transmission attempts varies, for different values of the average error probability over the wireless channel. The left plot shows the case where images are not compressed and $L = 2, 4$; the right plots presents the case of compressed images.