Combined Dynamic Power- and QoS-Management in Embedded Video Surveillance Systems

Arnold Maier[1], Bernhard Rinner[1], Helmut Schwabach[2]
and Thomas Trathnigg[1]

[1]Institute for Technical Informatics,
Graz University of Technology, A–8010 Graz, Austria
{maier,rinner,trathnigg}@iti.tugraz.at

[2]Video & Safety Technology,
ARC Seibersdorf Research, A–2444 Seibersdorf
{helmut.schwabach}@arcs.ac.at

Abstract — *Intelligent video surveillance (IVS) offers a large spectrum of different applications that have strict quality of service (QoS)-requirements. Beside high demands in computing performance, power efficiency is another major concern in these applications. For instance, recent IVS systems need to deliver compressed video data in high quality while using devices that are partly solar- or battery-powered. Since delivering high QoS-levels usually goes along with increased power consumption, it makes sense to implement combined QoS-management and power reduction methods. However, most IVS systems do have a lack of functionality in offering combined power- and QoS-management mechanisms.*

In this paper, we present PoQoS, a new generic concept for combined power- and QoS-management in distributed IVS systems. Its basic idea is to offer an extensible model that eases to implement coexisting dynamic power- and QoS-management mechanisms. Furthermore, our approach increases the systems reliability by including parameters of its actual energy state as well as information about fault recognition. Experimental results demonstrate the feasibility of our approach.

Keywords: *Intelligent video surveillance; Power aware distributed systems; Compressed video streaming; QoS; Dynamic power management; PoQoS*

1 Introduction

Over the last years, intelligent video surveillance (IVS) has become an important research area due to its broad field of different applications. IVS combines video- sensing, data-compression and analysis as well as short- or long term storage in order to yield certain parameters of the monitored scene. A lot of these applications have demanding requirements in delivering high level QoS while having strict energy constraints as well. High QoS-requirements in video data processing are typically satisfiable by using highly performance efficient hard- and software components. However, the use of such components often goes along with a rising amount of power consumed making power consumption a serious design constraint in IVS as well.

Beside of being a trendy term, power awareness has therefore also become an important issue in the development of IVS systems. Generally, power awareness has been broadly discussed in the design of various embedded electronic systems throughout the last decade. Increased power consumption has been arising from ascending functionality, complexity and speed. Especially in IVS systems, lowering the power consumption[1] is beneficial due to several reasons as summarized in the following section.

1.1 Why Power Aware IVS Systems?

Lowering the power consumption sometimes is an inevitable assignment. Due to the lack of available power lines in exposed positions, parts of an IVS such as cameras often are only solar- or battery-powered. In this case, the limited amount of available energy minimizes the time of operation and endangers the delivered QoS to be degraded improperly. Furthermore, power aware system design helps to minimize size and weight of certain facilities since smaller batteries and solar cells can be used.

Low power consumption is not only important in portable but also in line powered devices. Since electric current results in waste heat, heat dissipation is another serious problem that appears. If a certain amount of power is consumed, additional cooling devices need to be installed in order to avoid thermal problems that even can lead to destruction. In safety-critical surveillance applications, none of the employed facilities may fail due to high demands on reliability. However, especially active cooling devices are an additional source of uncertainty in reliability.

Furthermore, malfunction caused by thermal problems leads to costs for maintenance and repair making power awareness an important economical factor as well.

Another aspect that makes power awareness important in video surveillance is image quality distortion caused by noise. Sophisticated surveillance cameras take use of CMOS sensors due to their high dynamic range and logarithmic behavior. However, the more power a camera consumes, the higher its temperature is. Since the sensors are quite sensitive to high temperatures, it results in a decreased signal-to-noise ratio. To keep the advantages of a CMOS sensor in high image quality, minimizing the power consumption is inevitable and allows that no additional cooling for the sensor needs to be used.

Last, but not least, minimizing the power consumption is an important environmental and practical aspect. Beside of saving power, the environment is less affected because no additional power lines need to be installed. Furthermore power awareness goes along with some practical aspects since it is not always feasible to have power lines on location. Fig. 1 summarizes the main reasons that indicate the use of power aware IVS systems.

1.2 Contribution and Outline

In order to minimize the power consumption of an IVS system, we extend the basic concept of dynamic power management (DPM) by also taking QoS-parameters into account.

---

[1] It is important to realize that *power* is not the same as *energy*. Energy is the time integral of power. A reduction of power does not always go along with a reduction of energy if for instance the device's execution time is prolonged due to decreased speed. However, for reasons of simplicity and because of common use, we continue to use the term *power* synonym to *energy* throughout this work.

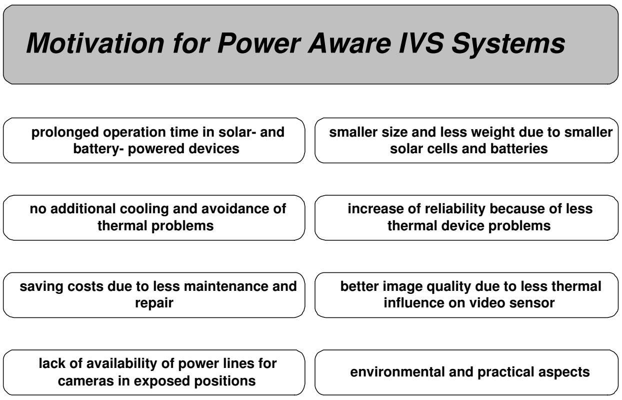

Figure 1: Motivation for Power Awareness in IVS Systems

Furthermore, we extend the model of DPM for its use in a distributed IVS. Therefore, we present PoQoS, a generic and extensible concept for combined power- and QoS- (so called 'PoQoS-') management in distributed IVS systems. Our approach combines distributed power- and QoS-control by using global and local adaptation strategies.

The remainder of this paper is organized as follows. In section 2, we give an introduction to IVS and its main applications. The architecture of an existing IVS system and its most relevant used components are illustrated as well. Section 3 gives a brief overview about related work on basic power reduction approaches. It presents the main ideas behind DPM, one of the most employed power minimization methods. Furthermore, we discuss QoS-parameters that influence power consumption in video encoding. In the next section, we present PoQoS, a generic concept for combined power- and QoS-adaptation in IVS systems. Section 5 reports some results of our current experimental work. The last section finally concludes and describes our future research activities.

2 Intelligent Video Surveillance (IVS)

Video surveillance is an area with widely spread different application areas. In this work, we focus on 3^{rd} generation *intelligent video surveillance* (IVS) as described in [1]. IVS is also based on the recent employment of intelligent video sensors [2], [3], [4]. These sensors combine video sensing with image processing and data communication. The design of the processing unit allows to yield various parameters of a captured scene and to compress a live video-stream simultaneously.

2.1 Application Scenarios

There exist diverse application scenarios for IVS, including safety critical applications such as traffic surveillance [5].

In traffic surveillance, existing conventional analogue cameras can be improved by additional processing units or even replaced by intelligent video sensors for better supervision of, e.g., tunnels or construction sites. This allows the recognition of dangerous situations and the generation of alarm signals to avoid consecutive endangerment of the situation. Fig. 2 shows an example for *stationary vehicle detection SVD* [6] due to an accident that occurred in a tunnel.

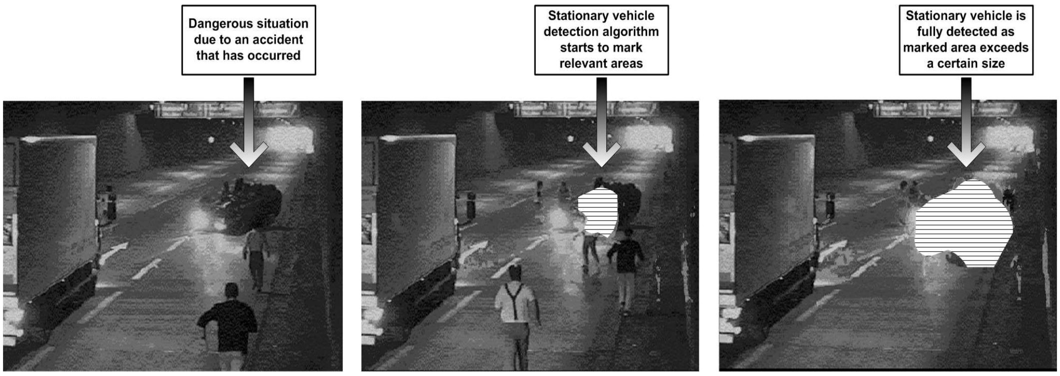

Figure 2: Stationary Vehicle Detection in IVS Systems.

The picture on the left side shows the dangerous situation some seconds after the accident happened. As it can be seen in this picture, a truck has also just stopped behind the car crash in this moment.

The next picture in the middle shows the monitored scene some frames after the left picture. The SVD algorithm starts to mark relevant areas in the monitored scene.

Again, some more frames later, the SVD algorithm has already fully detected the accident as the marked area exceeds a predefined size as shown in the right picture. However, since the truck only arrived a while after the car had already stopped, the SVD algorithm did not yet detect the stop of the truck.

It can also deliver valuable data for optimizing traffic light control in urban areas. Further applications [7] of IVS include the surveillance of buildings, persons or container shipping.

2.2 QoS in IVS Systems

It can be clearly derived from the above illustrated applications that QoS is an important consideration in IVS systems. Typical QoS-parameters are video data quality and its distortions in network transmission. As we often deal with compressed video data, these parameters include user perceived quality metrics such as the number of frames per second (fps), the image size, data rate or blockiness [8].

However, in IVS further quality parameters of application specific features are often taken into account. These include the availability and quality of additional services such as processed video parameter extraction like, e.g., stationary vehicle detection.

These applications usually have high demands in processing performance. For instance, video transmission at full PAL resolution and at 25 fps requires of approximately 20MB/s[2]. State-of-the-art video compression reduces the bandwidth needs by 2 orders of magnitude down to 1.5MB/s. The MPEG-4 encoding (advanced simple profile) for instance is well suited for this purpose in IVS systems.

In IVS systems, QoS is also linked with power parameters. In case of low energy in parts of the system, the QoS can also get seriously affected and degraded. If proper combined power- and QoS-adaptation is accomplished early enough, an all-out breakdown can be avoided if no recharging of available the energy storage is possible.

---

[2] Assuming 4:2:0 video format and 8 bits per pixel ADC resolution

2.3 The Digital Video System (DVS)

The DVS is a system for intelligent video surveillance and has been developed by the *Austrian Research Centers Seibersdorf* and *PKE Electronics*. The DVS is used for several applications in the field of IVS. It is based upon an open and scalable architecture that is to a high degree customizable and configurable due to its dedicated application. Its basic architecture is shown in Fig. 3.

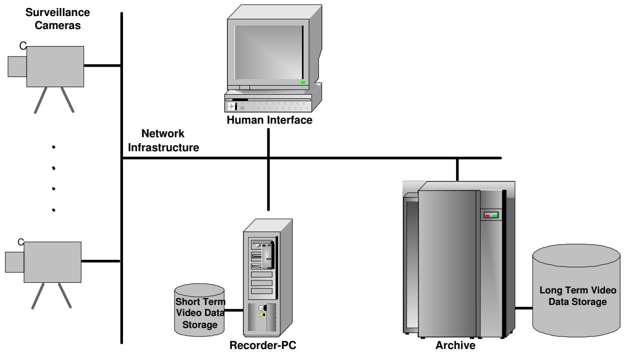

Figure 3: Architecture of the DVS

As depicted in the figure above, the DVS contains a variably number of cameras, whereas both conventional as well as intelligent cameras can get deployed. The video data is transmitted to a human interface that typically is located in a primary monitoring station. One of the main components of the DVS is the archiving unit that offers the possibility of long term storage. Depending on its implementation and size, it stores up to several hours of captured video data. This offers the possibility to document monitored scenes.

In an extensible number of so called recorder-PCs, additional features can be added to the DVS in order to extent its functionality. A recorder-PC contains extensible hard- and software for, e.g., additional digital video encoding or transcoding as well as video analysis. It also contains a circular video buffer that implements short term storage that for instance helps in retrieval of alarm situations.

For video data transmission and system control, the DVS uses a heterogenous network environment including wireless access for cameras in exposed positions.

3 Related Work

Minimizing the power consumption of an electronic system is an area of intense research. A lot of different power reduction approaches have been described in literature [9], [10]. Basically, two different approaches can be distinguished.

1. In *low-power design*, all methods and techniques are applied during design-time. It includes the selection of power efficient devices as well as power aware system design. Thus, beside of functionality, performance and costs, the parameter power is seen as an additional design criteria.
2. Beside of low-power design, further energy savings can be achieved by applying specific *online strategies*. These methods aim in reducing the power consumption

by selectively configuring hard- and software components during operating time due to the system's workload.

However, both low-power design and online strategies are typically based on same basic principles, that will be discussed in the next section.

3.1 Basic Power Reduction Principles

In an electronic system, it is often possible to do a trade-off between computing performance and power consumption. An example for trading performance for power is decreasing the system's supply voltage. This allows the performance of the system to be degraded. To maintain performance requirements, speed-up techniques such as parallel processing or pipelining need to be applied.

Another method to save power is to avoid waste caused by idle system modules. This approach aims in shutting down these modules. For instance, if a processor is not in use for a certain period of time, it is skilful to power down this component.

Other basic power minimizing principles include the exploitation of locality. In a distributed system, global data operations usually consume a lot of power. If, for instance, the same data is frequently transferred from one part of system to an other, the power consumption unnecessarily increases. On the other side, power is wasted if the same data is stored in different parts of the system. A design well partitioned to exploit locality minimizes the costs of global communication and consequently helps to save power.

Especially in distributed systems, choosing the most economic[3] communication medium is paramount. Adapting the system to application specific conditions is another way to lower the power consumption. For instance, in a distributed data processing system that computes varying input streams, dynamic reconfiguration like, e.g., workload balancing can help to save power. However, finding optimal partitioning strategies is a very difficult task to accomplish and typically needs to be developed off-line.

Computing data results in power dissipation due to the switching activity it causes. If the same data is processed frequently, it is better do a selective pre-computation in order to reduce the switching activity. However, this only leads to power reductions if the storage does not consume more power than the computation. Again, finding the optimal partition is a nontrivial problem.

3.2 Dynamic Power Management (DPM)

A commonly used and applied method is *Dynamic Power Management (DPM)* [11], [12]. It is based on several basic power reduction principles as described above. The goal of DPM is to minimize a system's power consumption without affecting the required performance. DPM is based on the observation that a lot of power is wasted because of system components that are unnecessarily and fully powered up even if they are not in use. Thus, the basic idea behind DPM is that individual components can be switched to different operating states (like, e.g., 'working', 'idle', 'sleeping' etc.) during runtime. Each operating state is characterized by a different set power- and performance- parameters.

Typically, the commands to change a component's operational state are issued by a central power manager. The commands are issued due to a corresponding power man-

---

[3] in terms of power consumption

agement policy. It usually is implemented in the operating system of the main processing component.

In order to decide which command to issue the power manager must have individual knowledge about the system's workload behavior. It also must take into account that changing a components operational state takes a specific time leading to latency of the device. Beside this, powering down and waking up again a device also consumes a specific amount of energy. Obviously, the more operational states exist, the better the systems operation can be tailored to its actual performance requirements by DPM.

A widely spread misconception about DPM is that it inevitably affects the system performance. However, if it is applied properly, it doesn't affect user perceived performance metrics.

Most of the applications described in the publications about DPM are for non-safety critical or general purpose systems with with non- or soft-realtime requirements. Safety critical or hard real-time applications often deal with combined problems of timing constraints and uncertainty, making DPM hard to apply. Its applicability mainly relies on the quality of its workload prediction.

3.3 Power Savings through QoS-Adaptation

Another power saving approach is based in the adaptation of the QoS of an application.

For instance if video data is processed, the power consumed by the processing unit typically depends on the quality of the video data. The degree of freedom in adapting QoS-parameters strictly depends on its designated application. If it is acceptable for the user and non-critical in the context of the application, the level of QoS may be manipulated and power can be saved in this way. Thus, this method is a trade-off between the loss of quality and the reduction of power consumption.

In a non-safety critical multimedia application like, e.g., internet media streaming, the QoS might temporarily be degraded without seriously affecting the user's satisfaction and safety. However, in a safety critical application like, e.g., video surveillance of traffic control, a degradation of the QoS-level may be unacceptable when it appears during the capturing of an accident.

As IVS applications deal with video data, a typically trade-off in QoS that yields to higher power efficiency is lowering the frame rate. When observing persons in a house for instance, reducing the frame rate from 25fps to 12.5fps does not seriously affect the user's perceived QoS but significantly lowers the power consumption of the processing unit. However, in traffic surveillance, it is already quite problematic to do so. If MPEG encoding is used, another power saving method is to skip P- and B-frames or to vary the image capture size.

3.4 Combined Power- and QoS-Adaptation

Throughout the majority of publications in the area of power aware video surveillance, both DPM and QoS-adaptation are usually seen as separate methods to reduce the system's power consumption.

Especially in IVS, combined power- and QoS-adaptation would make sense since both parameters are strictly depending on each other. Furthermore, power consumption also influences the system's reliability and therefore also the QoS of an IVS system.

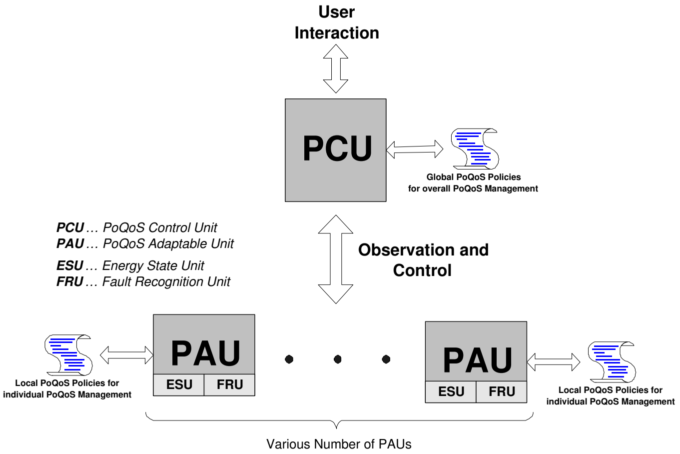

Figure 4: Basic Organization of PoQoS

However, only few publications exist about power aware video surveillance at all. In [13] for instance, the authors focus on the trade-off between energy consumption and image quality in wireless video surveillance networks. The power reduction is applied through QoS-adaptation whereas no hardware based DPM is applied.

In this work, we propose to apply combined dynamic power- and QoS-management in IVS systems as described in the next section.

4 Combined Power- and QoS-Management (PoQoS)

PoQoS is a new generic concept for combined dynamic power- and QoS-management in distributed IVS systems. Its basic idea is to offer an extensible model that eases to implement coexisting dynamic power- and QoS-management mechanisms. The intended distributed approach is build upon some ideas presented in [14] and is described in [15].

The infrastructure of an IVS system typically consists of a central monitoring unit that is connected to a various number of system units whose power- and QoS-level is adaptable dynamically. In PoQoS, all these units get abstracted due to their use for distributed dynamic power- and QoS-management.

Furthermore, PoQoS increases the system's reliability by including parameters of its actual energy state as well as information about fault recognition.

4.1 Architectural Concept

Fig. 4 illustrates the architectural concept of PoQoS that is applicable to many distributed IVS.

4.1.1 PoQoS Controller Unit (PCU)

The PCU implements the interface in between the user and the *PoQoS Adaptable Units (PAUs)*. A user interaction causes the PCU to issue commands to the corresponding PAUs due to global PoQoS policies. Beside conventional QoS management schemes such as network resource management for the overall system, there exist dedicated global PoQoS policies for different operation modes of the IVS system. Like that, proper PoQoS settings can be issued in various situations with different QoS-demands. For instance, in alarm situations it is usually necessary to deliver video data in best possible QoS. Furthermore, the global policies allow the implementation of additional power- and reliability- sensitive adaptation schemes as described later.

4.1.2 PoQoS Adaptable Units (PAUs)

A PAU can be any device in an IVS system whose PoQoS parameters are dynamically configurable. Examples of PAUs include devices such as video sensors, processing units or network devices.

Since a PAU's operation in a lower QoS-level usually leads to longer idle periods of its components, it makes sense to apply DPM as well. In PoQoS, each PAU employs its individual implementation of the DPM. Thus, a PAU contains its locally stored individual DPM policies for corresponding QoS-levels, i.e., it has its individual local PoQoS policies. Beside QoS-adaptation even higher power savings can be gained due to the coexisting employment of DPM.

A PAU also contains a local lookup table with a set of its predefined PoQoS levels. It lists the PAU's QoS-levels and their corresponding power consumption. Its purpose is to provide on demand information for the PCU. Obviously, the more PoQoS levels a PAU has, the better it is adaptable to actual requirements. The PAU also needs to deliver on demand status information to the PCU and to execute the PoQoS control commands issued by the PCU.

4.2 Communication Scheme

PoQoS is build upon a heterogenous network environment. Thus, it assumes as little as possible about the underlying network. It uses both event- and time-driven interaction schemes that work independent of the underlying network topology and communication protocol.

PoQoS allows the definition of different events for both PCU and PAUs. In the PCU for instance, user interaction causes an event. It stimulates an action like, e.g., to issue a PoQoS command by sending a message to a PAU. If the PCU or a PAU sends a message, the receiver needs to acknowledge its received content in order to avoid communication errors. Furthermore, the PCU uses a time-driven observation scheme for the PAUs in order to recognize malfunction or breakdown of a single unit.

PoQoS includes several mechanisms in order to increase the reliability of the system. Therefore, a PAU is specified to contain an *Energy State Unit (ESU)*. Especially in solar- or battery powered PAUs, it makes sense to report the remaining energy capacity to the PCU whether on demand or if it is below a critical threshold. Beside the ESU, a so called *Fault Recognition Unit (FRU)* detects malfunction of the PAU such as damaged cooling devices and sends a message to inform the PCU.

4.3 Typical PoQoS Workflow Scenario

For better comprehension of the functionality of PoQoS, we describe a typical PoQoS workflow scenario as illustrated in Fig. 5.

In this example, we consider a simple PoQoS system for intelligent traffic control that consists of a PCU and two PAUs, i.e., PAU_1 and PAU_2. Both units have various configurable PoQoS levels. PAU_1 is a solar powered video sensor without any available additional power line and thus with a limited amount of energy. It transmits video data to a central monitoring station that also contains the PCU. In our assumption, PAU_2 is high performance processor card that is located in a line powered computing rack in the monitoring station. PAU_2 is used for realtime video analysis of the scene captured by PAU_1. In our case it executes stationary vehicle detection. Due to its high level of heat dissipation, active cooling is used in PAU_2.

For reasons of simplicity, we assume correct inter-unit communication and thus omit communication error avoidance mechanisms as previously described.

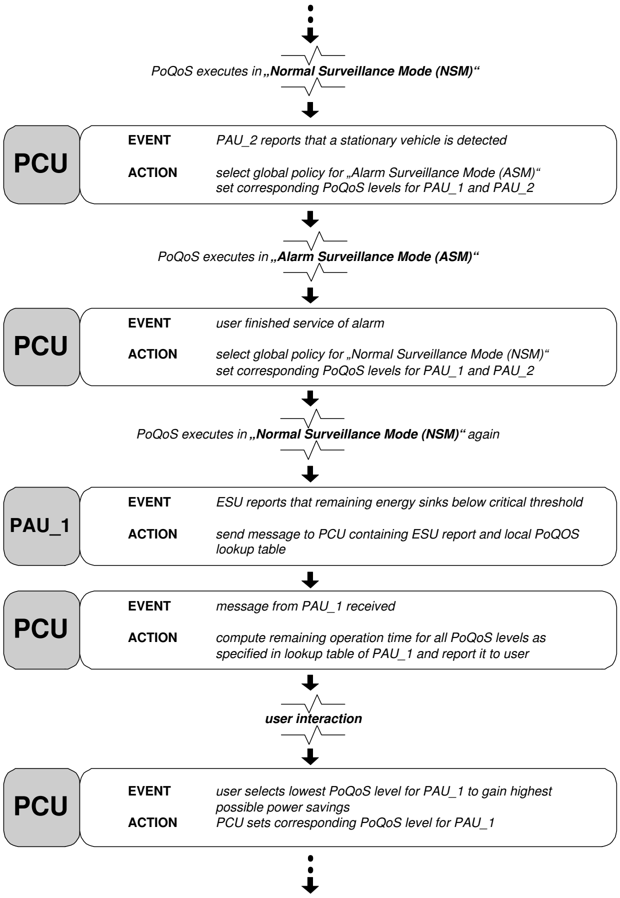

Figure 5: Typical PoQoS Workflow Scenario

In this example, we assume that PoQoS implements two global PoQoS policies, i.e., one for the so called *Normal Surveillance Mode (NSM)* and one for the *Alarm Surveillance Mode (ASM)*. After startup, the PCU chooses NSM as global PoQoS policy. This policy specifies proper PoQoS parameters for both PAUs. Thus, adequate local PoQoS policies are selected for each PAU. This allows to deliver video data in sufficient quality for the user and for the stationary vehicle detection without wasting power unnecessarily. After a

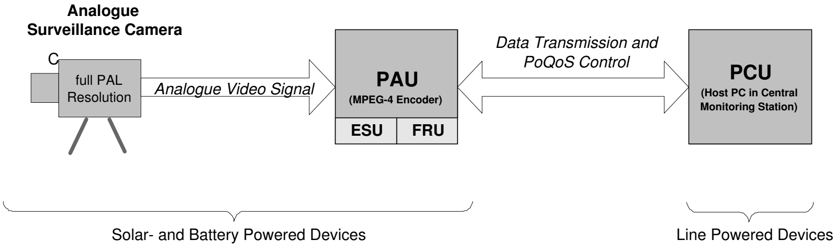

Figure 6: Block Diagram of the Experimental PoQoS Setup

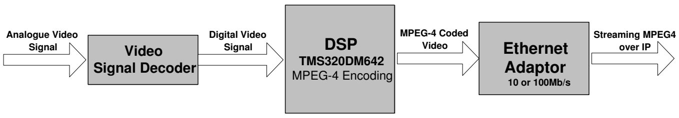

Figure 7: Functional Overview of the SCC

while, the stationary vehicle detection recognizes an alarm situation. As consequence, the PCU changes its global policy to ASM in order to choose the highest PoQoS parameters for both PAUs. Like that, a maximum QoS-level is provided to the user interface which is important in monitoring dangerous situations. After the alarm is served by the user, the PCU changes back to NSM. Again, after a certain time, the ESU of PAU_1 detects that the remaining energy is below a critical threshold. As a consequence, the PCU calculates the remaining operation times of PAU_1 for all its PoQoS levels[4] whereas a proper level can be chosen by the user.

5 Experimental Setup and Results

We evaluate the feasibility of PoQoS with a simple experimental setup that implements video sensing and encoding.

5.1 Experimental PoQoS Setup

Fig. 6 shows a raw block diagram of the experimental PoQoS configuration. As depicted in this figure, it contains a conventional camera that delivers an analogue video signal in full PAL resolution at 25fps. It is directly connected to a DSP-based MPEG-4 encoding hardware that was designed by the *Austrian Research Centers Seibersdorf*. The analogue camera does not offer the possibility to alter its PoQoS parameters. Thus, it cannot be used as PAU in this experimental setup.

5.1.1 Functional Overview of the PAU

Fig. 7 gives a functional overview of the Single Channel Codec (SCC) that is used as PAU in our setup. The SCC first captures the analogue video signal. The result of the capture phase is digital video data. MPEG-4 encoding (simple profile) is used to reduce network bandwidth needs to a maximum of about 1.5MB/s (in PAL resolution with 25fps). The MPEG-4 encoding is done via software using an encoding software from ATEME and gets

---

[4] as specified in its local lookup table

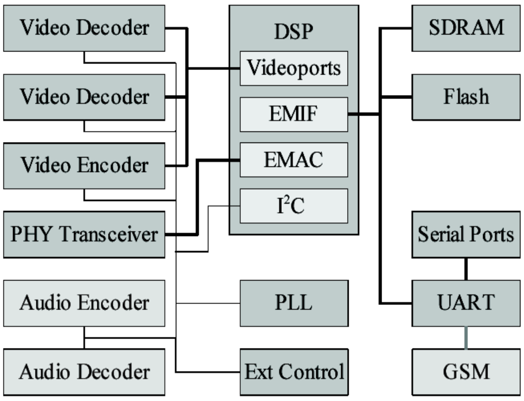

Figure 8: System Overview of the SCC

streamed via the network. The network connectivity is given by a TCP/IP stack provided by Texas Instruments, wheras real-time protocol (RTP) and multicast transmission is used.

5.1.2 System Structure of the SCC

Fig. 8 gives an overall system structure of the SCC. The SCC contains two video decoders and one video encoder and is capable of using composite video as input and output. Its main part is a DSP, whereas a TMS320DM642 from Texas Instruments is used. It provides three video ports, an external memory interface (EMIF) and an ethernet media access controller (EMAC). The video de/encoders are directly connected to the DSP. The same applies to the PHY transceiver and the SDRAM. Most of the peripherals are controlled vie the I^2C bus that is hosted by the DSP. The external control module can be used for controlling the orientation of the analogue camera or other input channels. In the given setup, the audio decoder and encoder are not used for reasons of simplicity. The optional GSM module provides wireless connectivity to the SCC, either in case of an ethernet failure or if no network cable can be used due to legal regulations. The onboard ethernet adaptor connects to the central monitoring station that also contains the PCU.

5.1.3 DPM in the SCC

In this implementation, DPM is integrated in the application framework on the DSP. The power manager is called upon every task switch and maintains a structure for every individual task containing all PMCs used by the corresponding task. Each PMC has several elements associated with, including a DPM policy, the actual power state and a mechanism for changing the power state.

The DSP can power-down mode its processor core by register control and can get woken up by predefined interrupt sources. Changing the DSP core's power mode only takes a few clock cycles. Thus, the effect of latency can get neglected. The video decoder chip also offers a power down mode that is controlled via I^2C. In contrast to the DSP, altering the power mode takes a varying amount of time that cannot be neglected. Therefore, the chip used in this implementation can only be powered down when lower frame rates are executed and longer idle times of the device are guaranteed.

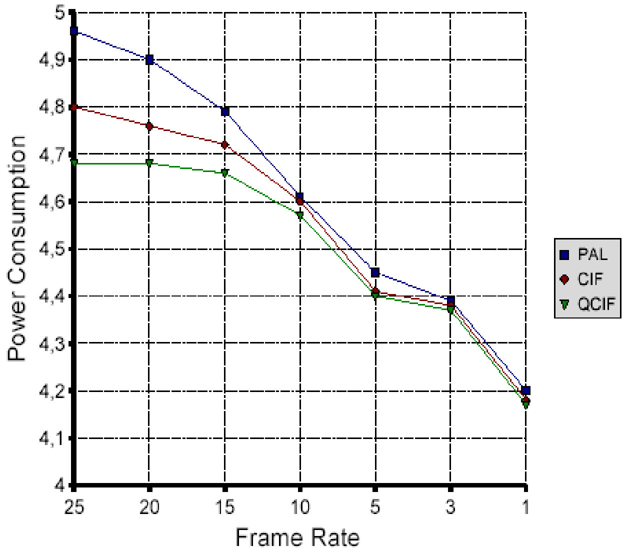

Figure 9: Power Consumption of the SCC with DPM of the DSP and Video Decoder

In a ”bursty” network transmission mode, also the PHY transceiver should enter its sleep mode. In the given setup, this would result in additional power savings up to 200mW. However, measurements have shown that the PHY device never enters its sleep mode. This may be caused by line activity, a hardware bug or the implementation of the TCP/IP stack that is used. Unfortunately, the reason stays revealed so far and, thus, the device is not used for DPM in this evaluation.

5.2 Results

The total power consumption of the SCC is measured by a digital oscilloscope. A current probe is used for measuring the current on the supply line. Furthermore, the supply voltage is also measured and the actual power consumption gets calculated on the fly by the oscilloscope. As the power consumption varies with the actual load of the system, the average over three seconds is taken. In the ’standard’ implementation (i.e., without applying DPM), the power consumption of the SCC is about 5.9W. Interestingly, the value is almost independent of the QoS-level due to the idle clocking activity of the DSP core and the video decoder even in lower QoS-levels. For our experimental setup, the power consumption has been measured under different PoQoS-levels (i.e., including DPM):

- PAL, CIF and QCIF resolution
- 25, 20, 15, 10, 5, 3 and 1fps

The results of the measurements with the SCC are shown in fig. 9. As seen in this figure, PoQoS yields even higher gains in lower QoS-levels. Especially in frame rates below 10 fps, the video decoder chip can also be powered down for a longer period of time without risking latency effects due to its previously described behavior. Thus, the SCC consumes about 40% more in its ’standard’ implementation than if PoQoS is applied.

6 Conclusion

In this work, we explained the necessity of power awareness in IVS systems. Since it makes sense to apply combined DPM and QoS-adaptation, we presented PoQoS, a extensible concept for combined dynamic power- and QoS-management in distributed IVS systems. Furthermore, PoQoS offers mechanisms that helps to increase the system's reliability. Experimental results illustrated the feasibility and showed the high power savings that are achieved by applying the PoQoS concept.

Future work includes the development and evaluation of global and local policies for the use of PoQoS in the DVS. For this purpose various experiments and measurements will be necessary. This includes the measurement and evaluation of the correlation of video data quality parameters and its corresponding power consumption of several PAUs of the DVS. Beside, latency of individual PAUs due to dynamic altering their PoQoS state needs to be taken into account as well. In finding optimized policy behavior for PoQoS, we will focus on adaptive parameter learning mechanisms similar to the ideas described in [16]. The idea is to implement an adaptive solution with pre-characterized PoQoS policies that get optimized by including online parameter learning.

Moreover, future work will include the extension of the given experimental setup by using more than one PAU. In this distributed setup, the PAUs also can communicate with each other. The PAUs then also can alter their individual policy behavior by user intended reconfiguration.

References

[1] C. S. Regazzoni, V. Ramesh, and G. L. Foresti, "Special issue on video communications, processing, and understanding for third generation surveillance systems," *Proceedings of the IEEE*, vol. 89, no. 10, pp. 1355–1367, Oct 2001.

[2] M. Bramberger, R. P. Pflugfelder, B. Rinner, H. Schwabach, and B. Strobl, "Intelligent traffic video sensor: Architecture and applications," in *Proceedings of the Telecommunications and Mobile Computing Workshop*, 2003.

[3] W. Wolf, B. Ozer, and T. Lv, "Smart cameras as embedded systems," *IEEE Computer*, pp. 48–53, 2002.

[4] M. Bramberger, A. Doblander, A. Maier, B. Rinner, and H. Schwabach, "Innovative video applications on high-performance dsps," in *Proceedings of the TI Developers Conference, Houston*, 2004.

[5] M. Bramberger, R. P. Pflugfelder, A. Maier, B. Rinner, B. Strobl, and H. Schwabach, "A smart camera for traffic surveillance," in *Proceedings of the First Workshop on Intelligent Solutions in Embedded Systems*, 2003, pp. 153–164.

[6] R. P. Pflugfelder and H. Bischof, "Learning spatiotemporal traffic behaviour and traffic patterns for unusual event detection," in *26th Workshop of the Austrian Association for Pattern Recognition*, 2002, pp. 125–133.

[7] G. L. Foresti, P. Mähönen, and C. S. Regazzoni, Eds., *Multimedia video-based surveillance systems*, Kluwer Academic Publishers, Boston, 2000.

[8] R. Steinmetz, Ed., *Multimedia-Technologie: Grundlagen, Systeme und Komponenten*, Springer Verlag, Heidelberg, 2000.

[9] M. Pedram and J. Rabaey, Eds., *Power Aware Design Methodologies*, Kluwer Academic Publishers, 2002.

[10] A. Maier, "Power awareness in embedded systems," *Technical Report TB 03/03, Institute for Technical Informatics, Graz University of Technology*, Aug 2003.

[11] A. Bogliolo L.Benini and G. De Micheli, "A survey of design techniques for system-level dynamic power management," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 8, no. 3, Jun 2000.

[12] IBM and MontaVista Software Inc., "Dynamic power management for embedded systems," *Technical Report Version 1.1*, Nov 2002.

[13] C.F. Chiasserini and E. Magli, "Energy consumption and image quality in wireless video-surveillance networks," in *Proceedings of the 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2002.

[14] L. Abeni and G. Buttazzo, "Hierarchical qos management for time sensitive applications," in *Proceedings of the Seventh IEEE Real-Time Technology and Applications Symposium*, 2001, pp. 63–72.

[15] T. Trathnigg, "Combined management of power and quality of service in video surveillance systems," M.S. thesis, Graz University of Technology, Graz, Austria, 2004.

[16] J.H. Lu and G. DeMicheli, "Comparing system-level power managing policies," *IEEE Design & Test of Computers*, Mar-Apr 2001.