Design of a Heterogeneous, Energy-Aware, Stereo-Vision Based Sensing Platform for Traffic Surveillance

U. A. Khan, M. Quaritsch, B. Rinner
Institute of Networked and Embedded Systems
Pervasive Computing Group
Alpen Adria Universität Klagenfurt
9020 Klagenfurt, Austria
Email: umair.khan@uni-klu.ac.at, markus.quaritsch@uni-klu.ac.at, bernhard.rinner@uni-klu.ac.at

***Abstract*—The biggest challenges faced by intelligent traffic monitoring systems are mobility, compactness and energy-efficiency. Current traffic monitoring systems are based on fixed installations and thus have no or least portability. Also, they use many sensors (e.g, cameras, induction loops, radar or laser), utilize little or no image processing capabilities, and are difficult to set-up. As images contain a lot of information, the surveillance systems purely based on vision can help avoiding the use of additional sensors, reducing the size of the sensor platform and hence increasing the flexibility and mobility. Since mobile systems often run from batteries, power consumption is a major issue and these systems should be highly energy-efficient. In this paper, we describe the heterogeneous sensor architecture of our mobile traffic surveillance system *MobiTrick* and its potential dynamic power management. The use of heterogeneous sensors is motivated by utilizing the 3D stereo information from the heterogeneous visual sensors to perform the required operations and thus avoiding the use of other large sensors.**

I. Introduction

Current traffic monitoring systems are commonly based on sensor nodes, each containing at least one (or more) camera that performs continuous or event-based scene capture, and other sensors, e.g., induction loops, laser scanners and radars, etc. These systems allow the implementation of various surveillance applications that may provide multiple services. These typical services include video streaming, incident detection, vehicle classification, or the computation of different traffic statistics [1]. The most common type of current systems is based on stationary installation where the sensor nodes are permanently mounted at gantries. Such fixed-mounted installations are usually expensive to set-up and decrease the flexibility of the monitoring system. Not only the road needs to be closed during the set-up time, but also a lot of calibration effort is needed. Portability is the major issue in such systems and in case of change in road infrastructure, a lot of effort is required to uninstall the system and deploy it back.

Therefore, a mobile and compact traffic monitoring system that can provide the typical traffic surveillance services is highly required. Mobile devices are more efficient in enforcement and can also react more flexibly to changing road situations such as construction sites. Such system should easily be transported from place to place but during operation it should remain stationary (for hours, days or weeks). For setup, the road does not need to be closed. However, this level of mobility and portability is ensured only when the sensor platform is of smallest possible size. This can only be achieved by selecting the sensors carefully and avoiding the use of large sensors which are infeasible for mobile devices. Since a mobile system is meant to frequently change its location and to operate in different places, a frequent effort for calibration and re-adjustment is also required. In order to avoid this tedious process, the mobile system must have capabilities to calibrate itself without any external input.

Mobile systems also impose a restriction of power consumption. As these systems run from batteries, the energy demand must be kept low which also limits the amount of computation that can be performed. Setting a trade-off between the energy efficiency and the computing efficiency is one of the major challenges which requires an online power reduction strategy that can optimize the overall power consumption during the system's operation.

The proposed architecture of our traffic surveillance system (*MobiTrick*) is mainly focused on portability and mobility. *MobiTrick* utilizes the image processing capabilities to perform all the required tasks, including vehicle detection and classification, over-height estimation, incident detection (just to name a few). The main feature of *MobiTrick* sensor node is a heterogeneous setup and a stereo configuration with different types of cameras. This stereo information is required to perform most of the tasks (e.g., over-height detection, vehicle classification, etc) and hence eliminates the need of using other large sensors, resulting in an overall small sensor-node size.

The following sections of this paper discuss the related work for Dynamic Power Managemnt (DPM) in surveillance systems, the specification of *MobiTrick* sensor platform, the system prototype and potential DPM.

II. Related Work on DPM

Power optimization techniques can be classified into (i) static, and (ii) dynamic techniques. Static techniques include synthesis and compilation for low-power and are applied at design time. Dynamic techniques use runtime behavior to reduce power consumption when the systems are serving light workloads or are idle [2]. The latter is known as Dynamic Power Management (DPM) that can be achieved by switching to different states of the system based on the current workload. A DPM policy is generally exposed to the following challenges and requirements.

- Enumeration of the available power states of different components (i.e., working, idle, sleep, deep sleep, OFF).
- Latency involved in switching to different power states.
- Determination of current and future workload of the system for changing its power states.
- Behavior of the applications running on the system and their respective workloads.

The literature review on dynamic power management provides various techniques used in different types of sensor networks (e.g, wireless sensor networks, visual sensor networks). In a broad domain, DPM techniques can be classified into stochastic [2][3][4][5][6][7] and deterministic [2][4][5][8][9][10] techniques.

Stochastic approaches make probabilistic assumptions (based on observations) about usage patterns and exploit the nature of the probability distribution to formulate an optimization problem, the solution of which derives the DPM strategy [3]. The main characteristics of these approaches is construction (or validation) of a mathematical model of the system that leads to a formulation of a stochastic optimization problem. A later step follows the creation of strategies to guide the system's power profile that achieves the highest power savings in the presence of the uncertainty related to the system's inputs. Most of the stochastic DPM modeling approaches are based on Markov models (or chains).

Deterministic techniques attempt to predict the energy usage behavior of a node in the future, typically based on the past history of usage patterns, and decide to change power states of the node accordingly. The rationale in all deterministic techniques is that of exploiting the correlation between the past history of the workload and its near future in order to make reliable predictions about future events. These policies predict the length of an idle CPU period before it starts. If an idle period is predicted to be longer than the *break-even time* (the minimum length of an idle period to save power), the device enters a low power state *(sleep)* right after it is idle. In order to predict the idle periods, several techniques have been introduced. One such technique uses adaptive learning trees [8] that encode the sequence of idle periods into tree nodes. This policy predicts the length of an idle period with finite-state machines similar to multi-bit branch prediction in microprocessors. If an idle period is predicted longer than the break-even period, the confidence level increases; otherwise, the confidence level decreases [2]. Some other techniques use well-known mathematical filters (e.g., average, moving average, exponential weighted average, least mean square) on the past history of the system's workload in order to predict the future workload.

Another wide area of research in DPM is Dynamic Voltage and Frequency Scaling (DVFS) which exploits either stochastic or predictive techniques. DVFS exploits the fact that the amount of energy required for a given workload is proportional to the square of the processor's supply voltage and the clock frequency [11]. Most of the times, the peak performance of a processor is not required and hence by reducing the clock frequency or the supply voltage of the processor, the energy consumption can be reduced, possibly at the expense of longer time to complete a specific operation. The overall motivation of DVFS is to reduce the clock frequency and the supply voltage in such a way that the processor can meet the deadline for executing a specific task. Several techniques have been proposed and evaluated in literature for predicting the processor's operating frequency and voltage at the operating system level [12][13][14][15][16].

Restricting only to the surveillance systems, we classify the existing DPM techniques into (i) low-power embedded designs, (ii) computational load vs. transmission power, (iii) based on Quality of Service (QoS), and (iv) multi-tier approaches.

A. Low-Power Embedded Designs

Low-power embedded designs for surveillance comprise low- or medium resolution image sensors integrated with an embedded computing platform (mostly ARM processor based or equipped with low-power DSPs) [17][18][19][20][21]. The key power management strategy in such systems is based on system-level dynamic power management because embedded systems provide good power management in terms of better control and access to various system components like system buses, memory, communication units and power modes of processor and other components. The power management policy in such systems is usually implemented at operating system level which deals with turning OFF/ON individual system components or changing their power modes based upon the occurrence of events, available resources or processing requirements.

B. Computational Load and Transmission Power Trade-off

Some techniques in the literature demonstrate that the goal of minimizing the total energy consumption can be reached through a reduction of the communication burden, since it requires more power than the one required for computation [22][23]. Hence, in order to reduce the transmission burden, one of the solutions is to compress the images/data on-board. This solution faces two major issues: (i) the computational power required for image compression is comparable to the transmission; (ii) the quality of the images must not be much degraded in order to preserve readability. Another important consideration is whether to select lossy or lossless algorithms. In [22], the compression rates, processing times, and energy consumption of different compression algorithms are studied. However, the performance of these algorithms significantly depends on the applications and the overall sensor setup/configuration and thus the results cannot be generalized. This approach can be implemented just as a part of the overall DPM strategy and requires a brief investigation of the power consumptions and processing load of different compression algorithms for a particular setup/configuration.

C. QoS Based DPM

Another approach studied in the literature is *"combined power- and QoS adaptation"* [1][24]. Since the power consumption for processing depends on the quality of video data (frames per second, resolution), the level of QoS may be manipulated and power can be saved in this way. Thus, this method is a trade-off between the quality of service and the resulting power consumption. This approach uses different DPM policies for individual components (video sensors, processing units, network devices) to dynamically change their power modes based on the required quality of service. The system runs in normal mode when no event is detected and hence delivers minimum quality of service at the expense of least power consumption. At the detection of an event (e.g., a traffic jam or stationary vehicle), adequate QoS parameters are selected for each component to deliver video data in sufficient quality.

D. Multi-Tier Approaches

In situations where the sensors and nodes have different capabilities and power requirements, it is feasible to design the same application by employing heterogeneous elements. The multi-tier approach studied in literature [25][26] makes use of multiple hierarchical levels of heterogeneous sensors. In such a setup, lower levels *(tiers)* contain low-power sensors that work as triggers for the higher level sensors at the detection of an event. By this, resource-constrained, low-power elements are employed to perform simpler tasks, while more capable, high-power elements take on more complex tasks. The literature shows that the multi-tier approach in heterogeneous setup can optimize power consumption and maximize network lifetime as compared to the single-tier approach.

The relevant literature reveals that most of the surveillance systems are designed for very limited number of tasks where they don't have a wide range of application workloads. In the same way, they focus only on a single DPM aspect (e.g., QoS based DPM technique does not take into consideration the other DPM techniques like DVFS) and do not investigate the potential benefits of other DPM techniques. Furthermore, the literature does not provide adequate information about the DPM in heterogeneous sensors setups for surveillance systems.

From the literature survey, it can also be concluded that the multi-tier approach is better than the single-tier approach in surveillance systems, in the sense that low-power elements are employed to perform simple tasks and the more capable elements come into action only when required. However, the approach presented in [25] does not investigate the potential benefits of other DPM methods like QoS based DPM, or DVFS.

III. *MobiTrick* PROJECT SPECIFICATIONS

MobiTrick project is aimed at developing a compact, autonomous and energy-efficient mobile traffic checking system utilizing the image processing capabilities. The system is intended to work in a heterogeneous setup, i.e., different types of high-resolution visual sensors including RGB, grayscale, infrared, HDR (High Dynamic Range) and some non-visual sensors like Inertial Measurement Units (IMUs) and GPS receiver. The advantage of using heterogeneous sensors is many-folds; (i) distributing tasks among different sensors (e.g., license plate detection with an infrared camera and a context image with a color camera), (ii) performing low-level operations with less capable (and more power efficient) sensors and complex operations with more capable sensors, (iii) performing 3D measurments with heterogeneous sensors required for many tasks, (iv) exploiting the redundancy to increase reliability (v) avoiding the use of additional sensors, such as laser or radar.

The system is aimed to work in a heterogeneous stereo setup where the stereo information is used to perform most of the required tasks (e.g., vehicle detection, vehicle dimensions calculation). As the visual sensors perform the task of scene capture, non-visual sensors (IMUs and GPS receiver) are required to accurately measure sensor platform's position, tilt and vibration. In order to avoid the task of frequent calibration in this mobile system, we use auto-calibration techniques with which the system can adapt to the changing environments just by using the information that is readily available in the scene. Additionally, the system uses online learning techniques for the adaptive training of detectors (for robust vehicle detection according to the changing environments) by using 3D information from the scenes.

From the energy-efficiency perspective, we use a multi-tier architecture where we put the heterogeneous sensors into different levels according to their capabilities. This approach is based on the idea that all the sensors should not be operational all the time, but must be put into action only when required. In the lowest level, we use a low-power smart camera that can run on-board algorithms (e.g., vehicle detection) and can work as a trigger for the cameras at higher levels to start working. The cameras at higher levels work in a triggered mode, where they start scene capture only when they receive a trigger signal from the low-level camera. The trigger serves two purposes;

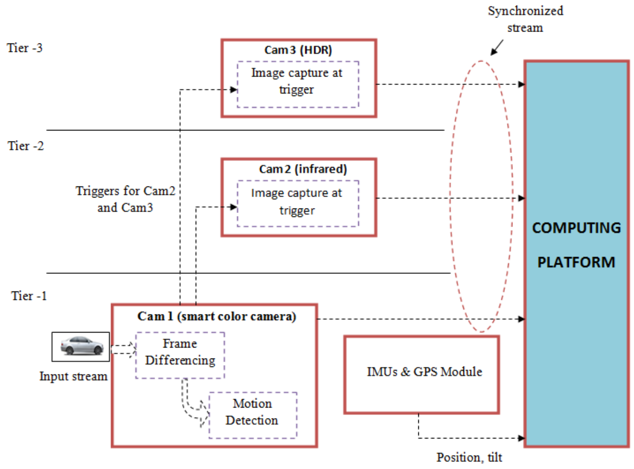

Fig. 1. *MobiTrick* heterogeneous sensor architecture

(i) invokes actions on the higher-level cameras, (ii) works as an input to the DPM policy running on the processing unit of the sensor node. The high-level architecture of our system is depicted in Fig. 1 where the higher-level cameras receive triggers from the smart camera running at the lowest level. Upon reception of these triggers, the cameras can then synchronize with each other and send the synchronized stream to the computing platform where this stream is processed to obtain 3D stereo information and to perform other tasks.

IV. System Prototype

While selecting the hardware component of the sensor node, energy consumption of each component was taken to be the major parameter. This turns out to be a challenging issue when desired functionality along with several important features are required. Besides many essential features of the visual sensors (e.g., frame rate, gain, noise, dynamic range, image sensor type, and mounting type, etc), resolution was given a special consideration in order to cover a larger field of view of the lane for detecting vehicles. A low-power, high-resolution smart camera, capable of performing different on-board tasks (e.g., motion detection, vehicle detection) was selected and put at the lowest level to work as an event monitor, where it remains in an always-operational mode and sends triggers to more capable cameras at higher levels. The current low-power design of our proposed system comprises two high-resolution but power-efficient cameras each having different features (HD smart RGB camera capable of streaming H.264 and MPEG-4 encoded video up to 1080p resolution, gray-scale Near InfraRed (NIR) camera). The NIR camera is useful for capturing images under poor light conditions to perform different tasks (e.g., license plate detection). Our prototype is intended to include a third camera also which would be a HDR camera to more accurately represent the range of intensity levels in the images. For the high-performance and energy-efficient computing, we use *ZOTAC ION-ITX* computing platform having Intel Atom N330, dual-core, 1.6 GHz processor with *NVIDIA ION* Graphic Processing Unit (GPU). For high-end computing, we are utilizing *NVIDIA's* parallel processing architecture *CUDA* for fast decoding, encoding, compression and execution of various vision based algorithms. Although the selected hardware (i.e., Intel Atom and NVIDIA GPU) is not commonly used in embedded devices, the power consumption is rather low while providing sufficient computing performance for image processing. Moreover, there is no embedded platform till present that uses a CUDA capable GPU.
Our preliminary experiments on video decoding and encoding using GPU (on our computing platform) show that the GPU-aided decoding and encoding not only off-loads the tasks from the CPU and is much faster than the one performed by CPU but is also more energy efficient than the CPU-based decoding and encoding. For H.264 and MPEG-4 video decoding (from the HD camera) to different formats, we use *NVIDIA's* VDPAU *(Video Decoding and Presentation API for Unix)* library [27] and achieve an average speed-up of 4x. The average CPU utilization with VDPAU is only 35-40% (which is almost 90% without VDPAU). For encoding to different formats, the average speed-up of 1.5x is reported.
In order to continuously measure the sensor node's position and orientation, we selected small-size and highly power-efficient IMUs and GPS module. Our selected IMUs are a low-power embedded sensor board *ArduIMU* (3-axis accelerometer, 3-axis gyroscope, Atmega328 processor) integrated with *U-Blox* GPS module. The power consumption features of different components on the sensor node are presented in Table I.

TABLE I
Power consumtion of different components on the sensor node

Components	**Max. power consumtion**
PhotonFocus camera (gray scale,NIR CMOS sensor, 1.4MP)	2.41 W
DMVA2IPNC camera (HD smart cam, 5MP)	2.00 W
ArduIMU sensor board	450 mW
Ublox-6 GPS MODULE	300 mW
NVIDIA ION GPU	6.00 W
Intel Atom N330	9.50 W
Computing board (idle)	23.00 W

V. MobiTrick Potential DPM

Unlike other DPM techniques that focus only on a single DPM policy, we aim to consider two important factors in a multi-tier setup: (i) dynamic voltage and frequency scaling, and (ii) quality of service. Since Intel Atom N330 does not have Intel's *SpeedStep* technology, no available Linux governors or tools like *cpufreq* can be used to dynamically adjust processor's frequency and voltage. Therefore, Intel Atom N330 always runs at maximum frequency and voltage and thus consumes maximum power even in idle state. However, we found that the processor does contain 8 throttling states in each of which it works with different frequency and voltage settings which can be manually adjusted. As an initial experiment, we look into the processor's workload periodically using Linux *vmstat* utility. Based on the current workload, we issue the appropriate signal to change the processor's state or keep it same.

Despite of using high-resolution cameras, the maximum resolution and frame rate is not always required. Therefore, another consideration for dynamic power management is the required quality of service. We decided to associate certain QoS parameters for individual applications, so that the appropriate quality of service is guaranteed along with the optimized power consumption. For instance, vehicle detection can be performed at the lowest tier with minimum resolution, frame rate and certain clock frequency of the processor. However, the complex operations triggered by the vehicle detection (e.g., vehicle classification or speed calculation) and performed by the cameras at higher tiers require higher resolution, higher frame rate and higher processor frequency. Consequently, at present we recognize three QoS parameters for individual applications; (i) resolution (ii) frame rate (iii) clock frequency.

Apart from these QoS parameters, we still have to identify the power states of individual components (e.g., processor, GPU, memory, hard drive, GigE interface, etc). Dynamic switching of these power states is strongly correlated to the QoS parameters. When lower or minimum QoS is required, the individual components may be switched to the lower power states. Therefore, our required DPM policy should not only dynamically adjust the QoS parameters, but should also change the power modes of individual components.

The prototype of our proposed system with an abstraction of our required DPM policy is depicted in Fig.2 where the three cameras are connected to the computing board with GigE interface. The *ArduIMU* sensor board and *Ublox* GPS module are connected to the computing board via USB cable. Since *DMVA2IPNC* camera has the least power consumption and runs an on-board vehicle detection algorithm, it is used at the lowest tier in free running mode. At the detection of an event, the camera sends a trigger as an RTSP signal over the network for the camera either at tier-2 or at tier-3 (based on the requirement). The same signal/trigger is received by the computing board where the power management module sets the appropriate QoS parameters, power modes for individual components and the frequency for the processor.
At the reception of the trigger, the cameras at higher levels are synchronized with the *DMVA2IPNC* camera and stream stereo images to the computing board. Since the images from the *DMVA2IPNC* camera are encoded in H.264 or MPEG-4 format, they are first decoded with GPU aid using *NVIDIA's VDPAU* library. The required vision algorithms are then applied to the images (with full or partial implementation on *CUDA*) and the processed images are encoded with GPU aid in an appropriate format and sent to a control station (or stored for later reporting). After the processing of an event, the appropriate QoS parameters, power modes and the processor's frequency are reset by the power management module.

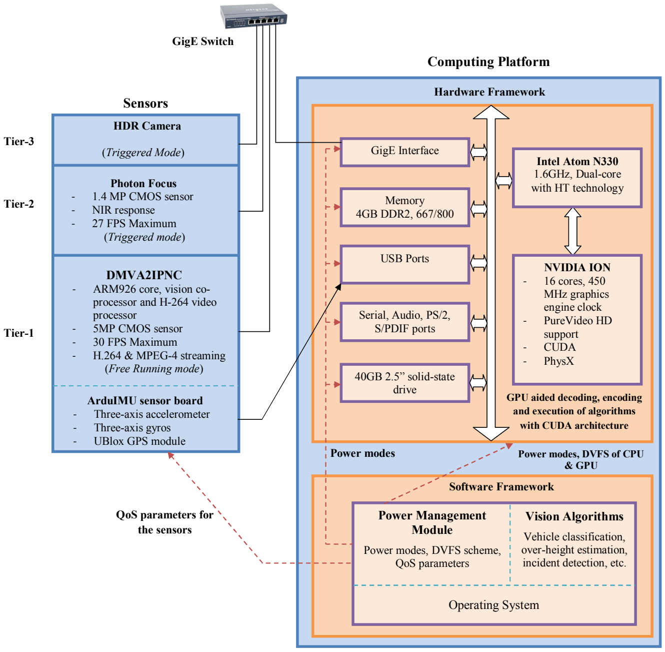
GigE Switch
Computing Platform
Hardware Framework
Sensors
Tier-3
Tier-2
Tier-1
HDR Camera
(Triggered Mode)
Photon Focus
- 1.4 MP CMOS sensor
- NIR response
- 27 FPS Maximum
(Triggered mode)
DMVA2IPNC
- ARM926 core, vision co-processor and H-264 video processor
- 5MP CMOS sensor
- 30 FPS Maximum
- H.264 & MPEG-4 streaming
(Free Running mode)
ArduIMU sensor board
- Three-axis accelerometer
- Three-axis gyros
- UBlox GPS module
GigE Interface
Memory
4GB DDR2, 667/800
USB Ports
Serial, Audio, PS/2, S/PDIF ports
40GB 2.5" solid-state drive
Intel Atom N330
1.6GHz, Dual-core with HT technology
NVIDIA ION
- 16 cores, 450 MHz graphics engine clock
- PureVideo HD support
- CUDA
- PhysX
GPU aided decoding, encoding and execution of algorithms with CUDA architecture
Power modes
Power modes, DVFS of CPU & GPU
Software Framework
QoS parameters for the sensors
Power Management Module
Power modes, DVFS scheme, QoS parameters
Vision Algorithms
Vehicle classification, over-height estimation, incident detection, etc.
Operating System

Fig. 2. *MobiTrick* sensor platform architecture

Our proposed DPM approach augments the multi-tier and QoS-based DPM approaches presented in Section-II along with DVFS in order to achieve better power reduction. The QoS-based DPM approach described in Section-II is focused on a single-camera node. Whereas, in our case, we are using multiple, heterogeneous sensors. Therefore, it is immensely useful to integrate QoS-based approach with multi-tier approach, so that not only we can set the appropriate power modes and QoS parameters according to the service level, but can also utilize the heterogeneous nature of the sensors to put them in different levels where they perform different tasks and come into action only when required. This will provide an additional power saving. Additionally, DVFS will also serve as another power saving factor for the computing platform where the voltage and frequency of the CPU and GPU can be varied based on the occurence of events and the workload being served.

VI. Conclusion

In this paper, we presented the prototype of our heterogeneous, energy-aware, stereo-vision based sensing platform for traffic surveillance. We highlighted the advantages of a heterogeneous sensor platform for surveillance system. The use of a stereo-vision setup based on heterogeneous sensor architecture helps avoiding the use of additional large sensors and provides flexibility. We also presented a survey of existing DPM techniques (in general and specifically in surveillance systems) and described the potential DPM policy in *MobiTrick*. In later steps, we aim to work on identifying the power modes of different components on the sensor node and implementing a concrete power management policy for switching these power modes and dynamically scaling CPU frequency. Furthermore, we also have to accurately determine the QoS parameters for each application. Another important consideration will be given to measure the latency involved in switching various system states and its impact on the intended system's operation.

ACKNOWLEDGEMENT

This work has been sponsored in part by the Austrian Research Promotion Agency under grant 825840.

REFERENCES

[1] Arnold Maier, "Dynamic Power Aware Camera Configuration in Distributed Embedded Surveillance Clusters," Master's thesis, PhD Dissertation, Technical University Graz, 2006.

[2] Y. Lu, G. Micheli, "Comparing System-Level Power Management Policies," *IEEE Design & Test*, vol. 18, no. 2, pp. 10–19, 2001.

[3] A. Fallahi, E. Hossain, "QoS Provisioning in Wireless Sensor Networks," *IEEE journal of wireless communication*, vol. 14, pp. 40–49, 2007.

[4] A. Sinha, A. Chandrakasan, "Dynamic Power Management in Wireless Sensor Networks," *IEEE Design and Test of Computers*, vol. 18, pp. 62–74, 2001.

[5] L. Benini, A. Bogliolo, G. D. 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, pp. 299–316, 2000.

[6] G. Norman, D. Parker, M. Kwiatkowska, "Using Probabilistic Model Checking for Dynamic Power Management," *Formal aspects of computing*, vol. 17, no. 2, pp. 160–176, 2005.

[7] Q. Qiu, Q. Wu, M. Pedram, "Stochastic Modeling of a Power-Managed System - Construction and Optimization," *IEEE Transactions on computer-aided design of integrated circuits and systems*, vol. 20, no. 10, pp. 194–199, 1999.

[8] E. Y. Chung, L. Benini, G. D. Micheli, "Dynamic Power Management Using Adaptive Learning Tree," in *Proceedings of the IEEE/ACM international conference on Computer-aided design*, 1999.

[9] T. Simunic, L. Benini, P. Glynn, "Dynamic Power Management for Portable Systems," in *Proceedings of the 6th annual international conference on Mobile computing and networking, MobiCom*, 2000.

[10] C. Hwang, A. C. Wu, "A Predictive System Shutdown Method for Energy Saving of Event-Driven Computation," in *IEEE/ACM International Conference on Computer-Aided Design*, vol. 5, 1997.

[11] Mudge. T, Ann Arbor, "Power: A First-Class Architectural Design Constraint," *IEEE journal of computer*, vol. 34, no. 4, pp. 52–58, 2001.

[12] D. Grunwald, P. Levis, K. Farkas, "Policies for Dynamic Clock Scheduling," in *Proceedings of the 4th symposium on operating system design and implementation*, 2000.

[13] K. Flautner, T. Mudge, "Vertigo: Automatic Performance-Settings for Linux," in *Proceedings of the 5th symposium on operating systems design and implementation*, 2002.

[14] T. Pering, T. Burd, R. Brodersen, "The Simulation and Evaluation of Dynamic Voltage Scaling Algorithms," in *Proceedings of the international symposium on low power electronics and design*, 1998.

[15] K. Govil, E. Chan, H. Wasserman, "Comparing Algorithms for Dynamic Speed-Setting of a Low-Power CPU," in *Proceedings of the 1st annual international conference on mobile computing and networking*, 1995.

[16] M. Magno, A. Lanza, "Energy Aware Multimodal Embedded Video Surveillance," in *IEEE conference on VLSI system on chip*, 2010.

[17] B. Brock, K. Rajamani, "Dynamic Power Management for Embedded Systems," in *IEEE international conference on system-on-chip*, 2003.

[18] Zhi-Yan Cao, Zheng-Zhou Ji, Ming-Zeng Hu, "An Image Sensor Node for Wireless Sensor Networks," in *Proceedings of the International Conference on Information Technology: Coding and Computing*, 2005.

[19] M. Bramberger, R. P. Pflugfelder, B. Rinner, "A Smart Camera for Traffic Surveillance," in *Proceedings of the First Workshop on Intelligent Solutions in Embedded Systems*, 2003.

[20] C. B. Margi, R. Manduchi, K. Obraczka, "Energy Consumption Tradeoffs in Visual Sensor Networks," in *12th Brazillian Symposium on Computer Networks*, 2006.

[21] W. Feng, E. Kaiser, M. Baillif, "Panoptes: Scalable Low-Power Video Sensor Networking Technologies," *ACM Transactions on multimedia computing*, vol. 1, no. 2, pp. 151–167, 2005.

[22] L. Ferrigno, S. Marano, A. Pietrosanto, "Balancing Computational and Transmission Power Consumption in Wireless Image Sensor Networks," in *IEEE Conference on virtual environments, human-computer interfaces, and measurement systems*, 2005.

[23] V. Raghunathan, S. Park, M. B. Srivastava, "Energy-Aware Wireless Microsensor Networks," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 40 – 50, 2002.

[24] A. Maier, B. Rinner, T. Trathnigg, "Combined Dynamic Power and QoS Management in Embedded Video Surveillance Systems," in *Proceedings of the second workshop on intelligent solutions in embedded systems*, 2004.

[25] P. Kulkarni, D. Ganesan, "SensEye: A Multi-Tier Camera Sensor Network," in *Proceedings of the 13th annual ACM international conference on Multimedia*, 2005.

[26] J. Sorber, N. Banerjee, "Turducken: Heirarchical Power Management for Mobile Devices," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, 2005.

[27] W. Hintermaier, E. Steinbach, "A System Architecture for IP- Camera Based Driver Assistance Applications," in *IEEE Intelligent Vehicles Symposium*, 2010.