Energy Efficient SRAM FPGA based Wireless Vision Sensor Node: SENTIOF-CAM

Muhammad Imran, Khurram Shahzad, Naeem Ahmad, Mattias O'Nils, Najeem Lawal and Bengt Oelmann.

***Abstract*— Many Wireless Vision Sensor Networks (WVSNs) applications are characterized to have a low duty cycling. An individual wireless Vision Senor Node (VSN) in WVSN is required to complete the tasks as quickly as possible. The execution of the tasks can be speeded up by exploiting the inherited parallelism in the tasks by using a hardware platform such as FPGA. Traditionally SRAM FPGAs are considered to be inefficient for duty cycled applications. This paper presents a low complexity, energy efficient and reconfigurable VSN architecture based on SRAM FPGA by using a design matrix which includes tasks' partitioning, a low complexity background subtraction, bi-level coding and duty cycling. The proposed VSN, referred to as SENTIOF-CAM, has been implemented on a prototype board and energy values of different states are measured for three real applications. The comparison results with existing solutions show that the proposed architecture with SRAM FPGA can achieve energy reduction of up to a factor of 69 as compared to software VSN solutions and approximately similar energy values to that for the FLASH FPGA based VSN solutions. The lifetime based on measured energy values shows that for a sample period of 5 minutes, a 3.2 years lifetime can be achieved with a battery of 37.44 kJ energy. In addition to this, the proposed solution offers a generic architecture with a smaller design complexity on a hardware reconfigurable platform and offers easy adaptation for a number of applications**.

***Index Terms*— Wireless Vision Sensor Node, SRAM FPGA, Wireless Vision Sensor Networks, Architecture, Image coding.**

I. Introduction

WIRELESS Vision Sensor Networks (WVSNs) are becoming increasingly prevalent within the research community as well as in industry because of the reduced infrastructure and maintenance costs, ease of deployment, scalability and low power stand-alone solutions [1]. The recent advancement in technology has enabled the WVSN to be used for a number of potential applications including machine vision [2], environmental monitoring [3], smart home [4] and surveillance [5][7][8]. WVSNs are often comprised of many individual wireless Vision Sensor Nodes (VSNs) which can capture and process data. Depending on the requirements, a VSN can make decisions [6], or transmit data to a user/server for further analysis [2][5][9]. As compared to traditional vision monitoring systems, which use wired communication and wall power supply, VSNs are interconnected with each other by using a wireless link. In the presence of limited resources such as processing, memory and wireless bandwidth, a VSN is often expected to operate for a greater length of time on the available limited energy. In many scenarios i.e., too many nodes, hazardous environment, etc, the battery replacement/ recharging is not feasible. Hence, the lifetime is a critical issue in relation to VSN and has been extensively studied in the literature [10][11]. The typical lifetime of a VSN can vary from a few days to a few years, depending on the application requirement [12].

In order to extend the lifetime of a VSN, researchers generally employ two strategies [2][3][4][5][9][13]. In the first strategy, as shown in Fig. 1-(a), no local processing is performed on the VSN and raw data is transmitted to the server for processing. This strategy has a smaller design complexity and consumes a smaller processing energy. However, this strategy consumes a greater communication energy because of the large amount of data being transmitted [3][4]. In the second strategy, depicted in Fig. 1-(b), all the required vision tasks are performed on the VSN and only the final features are transmitted to the server for analysis [9][13]. This strategy consumes a smaller communication energy. However, it consumes a greater processing energy on the currently available software platforms and has a high design complexity with regards to the hardware platforms [2].

In comparison to the aforementioned two strategies, the balanced approach is to partition the processing load between the VSN and the server as shown in Fig. 1-(c). This approach assists in energy reduction and the design complexity on a hardware platform and offers a generic architecture for this type of machine vision system [5][2][3][14] which has the ability to classify objects by using binary data [30]. In this paper, the approach shown in Fig. 1 (c) has been employed for VSN implementation. In relation to implementation, a VSN can be realized by using software and/or hardware platforms. On a software platform, the design and development time is

Manuscript received Jan. 20, 2013. This work was supported by STC@MIUN Research Program, Mid Sweden University, Sweden.

The authors are with Department of Electronics Design, Mid Sweden University, Sweden. (phone: 060-148416; fax: 060-148456; e-mail: muhammad.imran, khurram.shahzad, naeem.ahmad, mattias.onils, najeem.lawal, bengt.oelmann @ miun.se).

smaller because of the availability of ready to use image processing libraries. However, traditional software platforms perform computation sequentially, often one job at a time and cannot efficiently handle the VSN's front end tasks which account for a huge amount of data [15]. For low power systems i.e. VSN, the requirement is to finish tasks as quickly as possible so that the platform can be switched to a low power state in order to conserve energy [16].

The faster processing can be achieved by exploiting the inherited a parallelism in the front end tasks by using platform which offers parallel computation. Hardware platforms i.e. Application Specific Integrated Circuits (ASICS) and (Field Programmable Gate Arrays) FPGAs offer such parallel computation. ASICs have low power solutions and high performance as compared to FPGAs. However, the NRE cost for a low volume product is high and ASIC solutions are also inflexible with regards to any modifications. On the other hand, FPGAs offers parallel computing but still retain programmability of software at a relatively low cost [15][17][18]. These characteristics make FPGA a good choice for embedded vision processing [15][17].

FPGA technology varies from vendor to vendor and their classification is based on the configuration method [19]. The major FPGA technologies include FLASH and SRAM based FPGAs. FLASH based FPGAs store their configuration in logic gates. There is no requirement to download the configuration at each power-up [18][20]. These characteristics make a FLASH based FPGA suitable for duty cycle applications. On the other hand, SRAM based FPGAs are volatile and require re-configuration for each power-up cycle from the non-volatile memory [19]. These characteristics contribute to greater configuration and sleep energy which prohibits the use of SRAM based FPGA for duty cycled applications [15][21]. The advantages of SRAM based FPGAs include greater variation in device sizes, greater support and high performance because of advanced process technology as compared to FLASH based FPGAs [5]. Therefore, it is necessary to investigate techniques for employing SRAM based FPGA in duty cycled WVSN applications. In this paper, our goal is to investigate the use of SRAM based FPGA for duty cycled WVSN applications. VSN architecture with SRAM based FPGA requires special consideration for the implementation of vision tasks because of the non-volatility and greater configuration time. The traditional background subtraction techniques i.e., recursive and non-recursive techniques are expensive [22] and require the generation of a background at each power-up cycle which contributes to extra energy when there is no change in the background. In this work, we also investigated and developed a low complexity and low power background subtraction technique for the SRAM FPGA based VSN. Following this, section II presents related work, section III provides experimental work, section IV describes the VSN architecture, section V discusses the results and section VI concludes the paper.

II. Related Work

A number of VSN implementation strategies have been proposed by researchers in order to reduce the processing and communication energy consumption. Gasparini *et al*. [5], used a bi-level CMOS vision sensor of 128×64 resolution to capture the images and then perform binary processing on a FLASH based FPGA. The authors proposed design principles for VSN in the context of a long-lifetime. However, no discussion is provided regarding the use of SRAM FPGA for their work. Kerhet *et al*. [8] proposed a VSN, MicrelEye, for cooperative video processing applications. The vision tasks were processed on the SRAM FPGA and a microcontroller whereas, for data transmission, a Bluetooth radio was used. The authors provided the VSN results for active duration but a discussion relating to the whole duty cycle is missing. Sánchez *et al*. [23] proposed a video sensor node, which uses two Digital Signal Processors (DSPs) for image processing tasks and an FPGA for controlling the interconnection and image data flow. The authors focus is on the efficiency of the VSN tasks. Casares *et al*. [24] presented a lightweight and resource efficient foreground object detection and tracking algorithm for WVSN applications. The authors used a CMOS image sensor for data capturing and a microprocessor with embedded Linux for processing. Bakkali *et al*. [7] processed the tasks with a

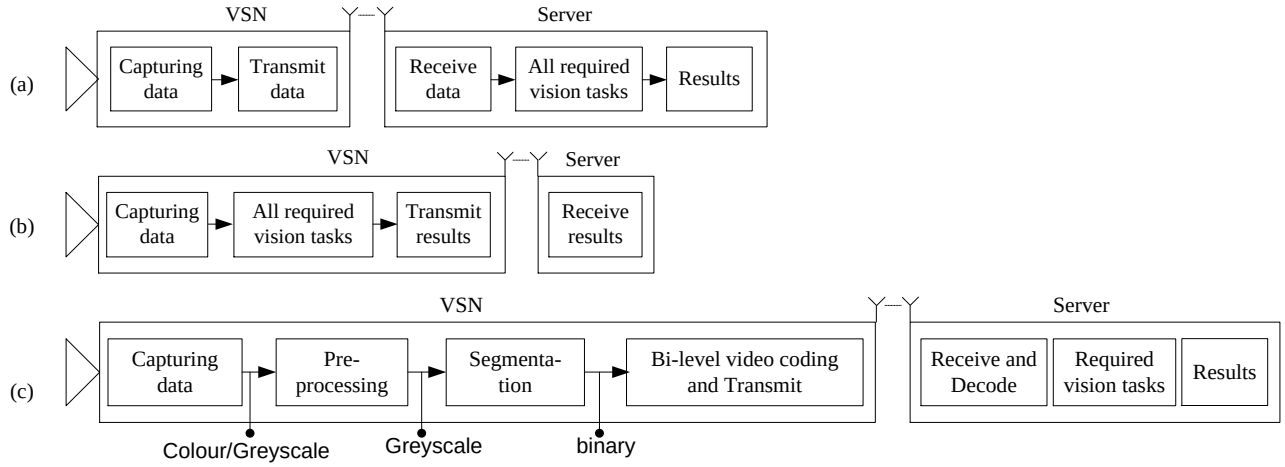


Fig. 1. VSN architecture, (a) with no vision tasks performed on VSN, (b all required vision tasks performed on VSN, (c) partitioning tasks between VSN and server.

regular computational flow on an array of processing elements and processed the tasks with small amounts of data on a 32 bit NIOS-II, RISC processor. The Stanford's MeshEye node [25] uses two kilopixel imagers for low resolution images and one high resolution camera module for capturing detailed object snapshots. MeshEye quantifies the reduction in energy consumption through the usage of a hybrid-resolution vision system. However, the hybrid vision system would suffer from calibration issues as variations in optics and alignment in image sensors would cause a problem in relation to object detection in the initial phase. Rowe *et al*. [13] presented a low-cost, open source embedded vision platform called CMUcam3 with their own optimized C library. The bottleneck in the CMUcam3 is based on the limited RAM and reduced computation speed, which is not sufficient for complex vision tasks. Kandhalu *et al*. [9] employed a platform called DSPcam which performs local processing in order to detect the event and will annotate the video stream for the operator in the observation station in the network. Juan *et al*. [21] investigated a SRAM based FPGA for VSN in the context of duty cycling but their investigation is not based on a real application environment. A discussion about the comparison of different VSN systems with respect to the proposed VSN can be found in the results sections.

In related work, a number of authors reported VSN systems with general purpose microprocessors, which are considered to be good in terms of flexibility and ease of use. However, the energy efficiency of microprocessor based systems is lower as compared to hardware implemented VSN systems [2][5][8]. The authors who used SRAM FPGA for VSN chose to report algorithms efficiency and energy consumption in the active state and the discussion about the transition states and sleep states is missing. To the best of our knowledge, our work is the first to consider the overall energy reduction in active, transition and sleep states for SRAM FPGA based VSN with real applications scenarios. This will assist in evaluating the lifetime of VSN in a realistic manner.

III. Experimental Setup

The following are the main components of the experimental work.

A. Applications description

The target area for the analysis of this work is machine vision applications in which objects can be classified by using binary data. For proof of concept, three test cases i.e., particle detection in an industrial machinery, remote meter reading and people counting have been used. All these cases represent machine vision applications in which the objects are changing slowly and the lighting conditions are controlled and for which an external LED flash light is used in order to achieve a high signal to noise ratio.

1) Industrial machine monitoring

Compared to a wired solution in industry, the wireless solutions offer many advantages such as flexibility in installing/upgrading the network, reduced deployment and maintenance cost, ease of re-location of devices, improved fault localization and isolation [26]. In this test case, we will consider an industrial application with a wireless requirement for the aforementioned reasons. It is important to mention that in some industrial applications, low energy may not be an issue because of the availability of a mains power supply. In industry, hydraulic machines wear out with aging and this can create a loss to industry due to accidental stoppages. Traditionally, the engineers stop the machine in order to check its health status by examining the oil. The oil becomes contaminated with magnetic particles, which detach from the engine with aging. The stopping of a machine will decrease the machine's productivity. The stopping can be circumvented by means of the introduction of wireless smart cameras, which continuously monitor the machines as shown in Fig. 2. The smart camera will automatically detect magnetic particles, measure their size and transmit the information to the user.

2) Remote water meter reading

In many countries, the electricity and water meters are electromechanical and replacing them with a digital option is expected to cost more than integrating a low cost wireless smart camera system. In some cases, the manufacturers/regulators do not allow the already installed digital meter's alteration in order to introduce direct integrated methods. In remote meter reading [3], a low cost and low power wireless smart camera is installed to monitor the meter and to transmit the information

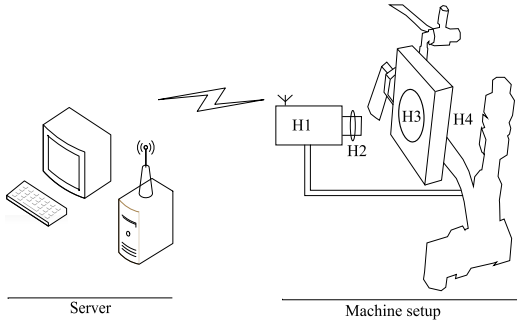


Fig. 2. Hydraulic machine setup with a server. (H1) VSN. (H2) LED ring. (H3) Window glass. (H4) Hydraulic machine.

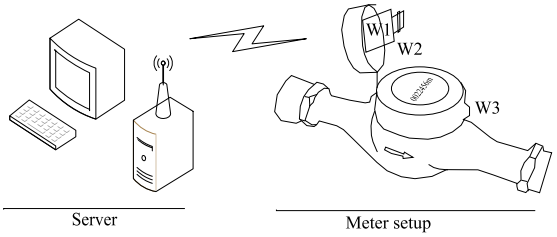


Fig. 3. Remote meter reading setup with a server. (W1) VSN. (W2) LED ring. (W3) Water meter.

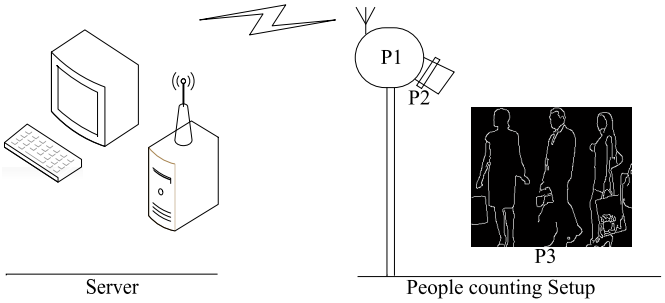


Fig. 4. People counting setup. (P1) VSN. (P2) LED ring. (P3) People passing.

regarding the counter digits to the receiver. This will assist the service providers in reducing the cost of hiring meter reading operators, will assist users to remotely monitor their energy consumption and will assist service providers to identify leakage points in the system. The setup for remote meter reading is shown in Fig. 3.

3) *People counting*

People counting are usually performed by using ray-trapping with fixed photocells and illuminators or by using techniques which involve PIR sensors, pressure sensors and thermal sensors. The battery operated, wireless camera based system offers another alternative with regards to people counting. This solution is compact and has the added advantage of surveillance over the other alternatives [5]. As discussed [5], this application is limited to people counting instead of full-fledged detection. The setup for people counting is shown in Fig. 4.

B. Processing platform

To realize a wireless VSN, the vision algorithms have been implemented on a SENTIOF platform [28] which is referred to as SENTIOF-CAM in this work. SENTIOF-CAM integrates components i.e. FPGA, micro-controller, SRAM, FLASH, CMOS sensor MT9V032 [29] and an IEEE 802.15.4 complaint transceiver in a single platform. The block diagram of SENTIOF-CAM is shown in Fig. 5. To receive the data at the server side, an IEEE 802.15.4 compliant transceiver embedded in a SENTIO32 [27] platform has been used.

C. Resource utilization, power and performance parameters

According to the tasks' partitioning approach [30], the tasks implemented on the VSN are shown in Fig. 6 whereas the tasks processed on the server are shown in Fig. 7. It is important to mention that on the server side, depending on the application requirements, the ordering and the types of tasks could be changed. The resource utilization of the individual VSN tasks is shown in Table 1. The resource utilization for the final implementation in which all the modules are integrated is given in section V. In relation to the final implementation, logic resources of any specific strategy can exceed the combination of the individual functions because the integration and synchronization requires extra logics. The processing time for each vision task is measured by means of a logic analyzer and can be calculated by using Eq.1

$$T = (Row \times (Col + Ls) + Lt)/\ f \qquad \text{(sec)} \qquad (1)$$

where *Lt* is the latency of each task, *f* is frequency, *Row* represents rows, *Col* represents columns and *Ls* represents low line sync. The power consumption of the VSN strategies is measured by using an Agilent 34410A meter [38] at a sampling frequency of 10 kHz. The time spent on transmitting the results to the server is measured on the actual hardware.

IV. SENTIOF-CAM ARCHITECTURE

The target architecture for this work is shown in Fig. 8 and has been implemented on the SENTIOF, which is discussed in section III-B. The front end tasks, background storage model and bi-level video coding have been implemented on an FPGA. The background storage in the FLASH memory is controlled by a state machine as shown in Fig. 9. After segmentation and morphology, the bi-level video coded data is transmitted to a server for further processing. It is important to mention that VSN tasks are the same and are fixed for different applications with the exception of the parameters i.e. image sizes, threshold values.

However on the server side, there is greater flexibility for incorporating different tasks and machine vision libraries. This tasks' partitioning approach assists in proposing a reconfigurable and generic architecture as the initial data intensive tasks can be easily reused on hardware reconfigurable platforms as compared to control dominated post segmentation tasks [14][31]. For conserving energy, the SENTIOF-CAM can be switched to a low power state, referred to as the sleep state, when the required vision tasks have been performed. In sleep state, only the real time counter is ON in order to keep track of the timing, whereas all other components i.e., FLASH, FPGA, SRAM and transceiver are OFF. In the proposed architecture, the minimum sleep duration should be 235 ms in order to effectively utilize the SRAM based FPGA for duty cycled VSN applications. This has been demonstrated in section V.

Following this, a brief detail of imaging tasks is presented.

A. Front end tasks

Front end tasks include image capturing, pre-processing, segmentation and binary morphology. The pre-processing includes the background subtraction and filtering.

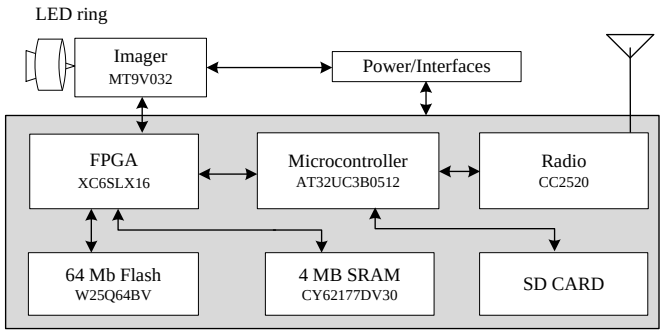


Fig. 5. SENTIOF-CAM block diagram.

TABLE 1. DEVICE UTILIZATION AND PERFORMANCE PARAMETERS OF VISION TASKS. AVAILABLE LOGICS 9112, BRAMS 64(8K*)/32(16K**).

Vision Tasks	Logics used	% used	BRAMs	% used	Latency (clk cycles)
Image capture	174	1.9	0	0	4
Background storage model*	691	8	1*, 2**	2*, 6 **	64†
Pre-processing	5	0.05	0	0	1
Segmentation	6	0.06	0	0	1
Morphology	139	2	4*	6	645
Change coding	209	2	3**	9	5
ROI	271	3	0	0	641
G4 Compression	3204	35	3*	5	647
Communication module [42]	100	1	1**	3	4

Note: † represent latency during first time configuration.

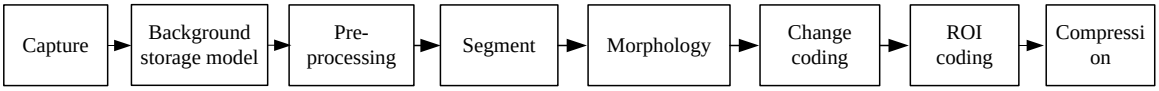


Fig. 6. Vision processing on VSN.

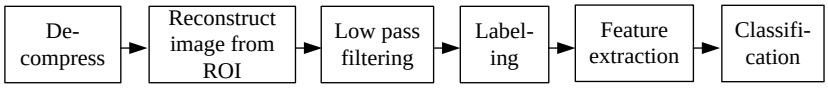


Fig. 7. Vision processing on server.

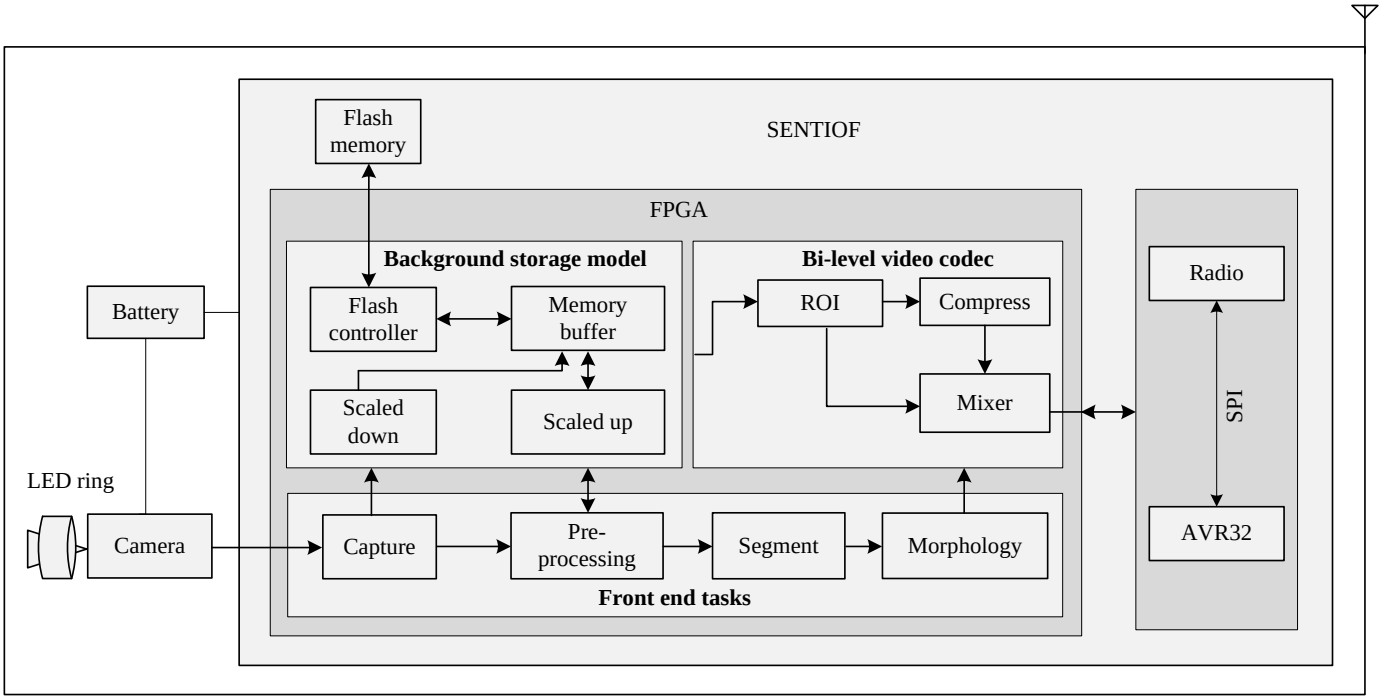


Fig. 8. Architecture of wireless vision sensor node, SENTIOF-CAM.

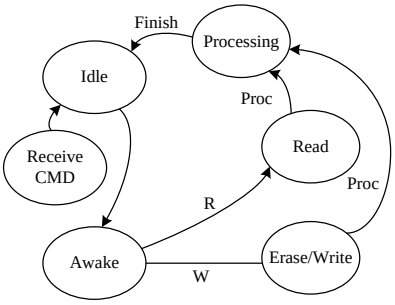


Fig. 9. State machine for duty cycling and background storage.

For background subtraction, the background image is stored in the FLASH memory [32] via a Serial Peripheral Interface (SPI) at an initial stage after which, the current image is subtracted from the background image. In relation to the people counting application, the pre-processing includes an edge detect filter. In this work, for proof of concept, we have used a sobel edge detect filter with a mask of 3×3. Following the pre-processing, segmentation is performed to partition the image into mutually exclusive connected binary regions. In binary morphology, erosion is followed by dilation, by using a suitable size of the structuring element. During the erosion, and dilation required number of rows are stored in order to have the necessary neighborhood information for the operation.

B. Low complexity background storage

In the context of a resource constrained SRAM FPGA based VSN, we propose a low complexity background subtraction technique, which uses existing image scaling techniques for image resizing in order to have smaller storage requirements. For the subtraction, operation, the downscaled version is upscaled by using an upscaling technique. Existing scaling methods include nearest neighbour, bilinear, bicubic, quadratic cubic, winscale, Lagrange and Gaussian [33][34]. Some of these techniques such as the Gaussian method, offer good quality but have higher computational complexity and require an approximately 22 times longer execution time as compared to that for the bilinear [33]. For the proposed background model, different image scaling techniques, including nearest neighbour, averaging, bilinear, and bicubic with different scaling factors were first investigated for a real application [35]. In relation to image scaling techniques, the output image quality, complexity and memory requirement depends on two

factors including scaling technique and scaling factor.

Therefore, we have considered these parameters in relation to selecting a suitable scaling technique for both upscaling and downscaling. Based on experimentations [35], it was concluded that averaging for downscaling and nearest neighbour for upscaling, with a scaling factor of 8, are suitable for the machine vision applications. The scaling factor of 8 means scaling in both width and height. It is important to mention that averaging during downscaling introduces blurring effect and nearest neighbour during upscaling might introduce artifacts. However, for images with a smooth texture, they both offer good results [34][35][36]. Depending on the requirements, spatial and anti-aliasing filters can be applied on the server side as shown in Fig. 7. These techniques have been investigated for indoor machine vision applications with controlled lighting and a smooth texture background. For applications with an outdoor environment, this background subtraction technique requires investigation. Nonetheless, the proposed technique will reduce the memory requirement by a factor of up to 64 in addition to a reduction in the design/implementation complexity as compared to the background model, which involves the storage of the whole frame. In this work, the proposed background storage and subtraction technique is implemented on hardware and the functionality is verified on the hardware. The background storage and subtraction model is shown in Fig. 10 and the state machine for controlling the background storage is shown in Fig. 9.

1) Background storage

For background image storing, the scaled down image by a factor of 8 is stored in the internal memory of the FPGA. The image size for two of the investigated applications, namely particle detection and meter reading, is 640×400. The image size for people counting is 640×320. The internal memory will lose its contents after power down. The contents that are lost, require the generating of the image after each wake up cycle, which is costly in terms of computation and energy consumption. It is important to mention that the background remains constant for a long time in many machine vision applications [5][35]. To handle this situation, the background scaled down image is stored in the non-volatile FLASH for later use. After each wake up cycle, the scaled down image is read out from the FLASH memory into the internal memory. If system parameters i.e. lighting, location, optics are changed, the background must be updated. For the subtraction operation, the scaled down image is accessed from the internal memory and is upscaled using a nearest neighbour technique. The background image reading, writing and erasing can be

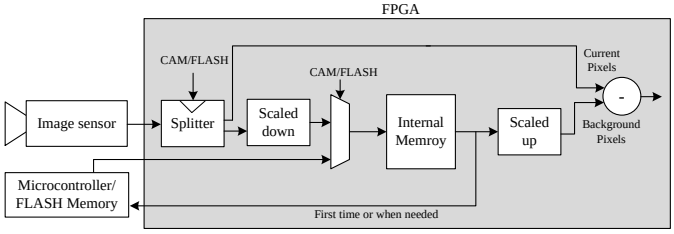


Fig. 10. Background storage and subtraction model with scaling.

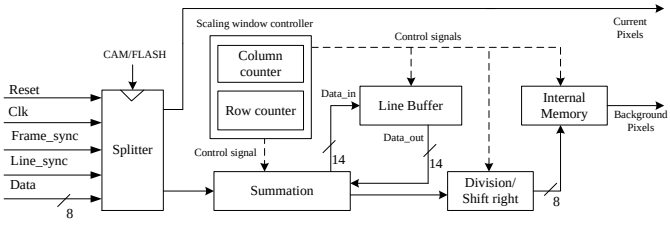


Fig. 11. Architecture for background image down scaling.

controlled by sending a command of size 1 byte from the microcontroller as shown in the state machine of Fig. 9.

2) RTL model of image down and up scaling

The architecture for image downscaling, using the averaging technique, is shown in Fig. 11. A scaling factor both for the width and height of the image can be set and controlled by *column* and *row* counters. A *line buffer* holds the intermediate summation values of the scaled down window pixels. For example, the sum value of a scaling window of 8×8 is stored in the line buffer in one location. When a specific scaling factor is reached in both the width and height directions, a signal is generated to *summation* and *division*/*shift right*, in order to perform division by right shifting the bits of the summed values and storing them in the internal memory of the FPGA. In the mean time, the pixels' values for the current scaling window will start summing. For upscaling, the nearest neighbour is used. This method requires the sampling of the nearest pixels to the original image [35] and the implementation is straightforward.

3) Background subtraction

For the background subtraction operation, the pixels from the background image are read out in advance in order to synchronize them with the current frame pixels for the subtraction operation. After the subtraction operation, the pixel data is forwarded for further processing.

C. Bi-level video coding

To reduce the amount of transmission data, we have developed and implemented a new bi-level video coding technique [30]. The architecture for this bi-level technique is shown in Fig. 12-(a) and the output data format is shown in Fig. 12-(b). The fundamental components of bi-level video coding include G4 compression, change coding, ROI coding and Huffman-run length code mixer. In G4, the coding scheme uses a two- dimensional line-by-line coding in which the position of each changing picture element, rather than alternating white and black runs in each scan line, is considered [37]. Change coding searches for the changes in consecutive frames. A change coded image is shown in Fig. 13-(e) for image sequences of Fig. 13-(c-d). The change coding removes similar objects. The ROI coding isolates those binary image regions which have objects. An ROI coded image is shown in Fig. 13-(f) for the image sequence of Fig. 13-(d). The run length codes, representing the presence of objects in the rows, are transmitted together with the Huffman codes of the ROI image to the server. The proposed video coding, comprising of the aforementioned components, has the flexibility to perform the following coding schemes.

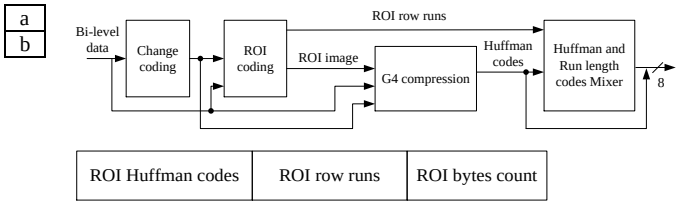


Fig. 12. Bi-level video coding, (a) architecture, (b) output data format.

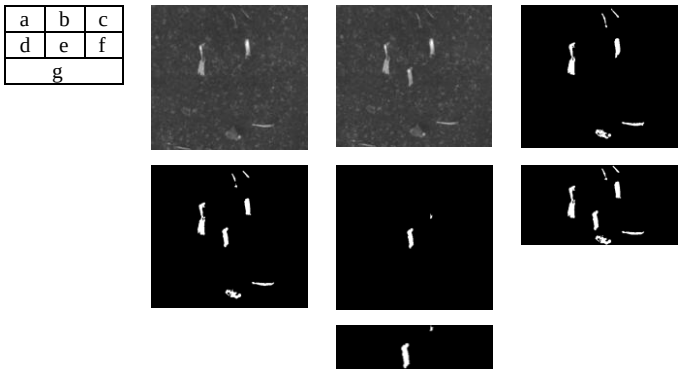


Fig. 13. Video coding example with images from particle detection application. (a)-(b) Image sequences. (c) Compressed image of *a*. (d) Compressed image of *b*. (e) Change coded image *a*,*b* (f) ROI coded image of *b*. (g) Change-ROI coded image of *a*, *b*.

- Image coding
- Change coding.
- ROI coding.
- Change-ROI coding.

Depending on the application and environmental factors i.e., the speed of the object and the lighting, the output data produced by the four coding schemes could be different and the coding should select the smaller bit stream based on the four schemes. Change coding requires the storage of the compressed frame in the non-volatile memory for duty cycling. The storage of a run length coded frame requires additional delay and energy. The additional delay is associated with sector erasing and page programming. For page programming, sector erasing is required. The typical time for sector erasing is approximately 30 ms for 4 KB. Each page programming in the FLASH memory requires typically 0.7 ms [32]. This time is greater as compared to the transmission of data with ROI coding. Therefore, in this work video coding with ROI coding has been selected.

V. Results And Discussion

The proposed VSN architecture is implemented on real hardware and its functionality is verified for three applications, namely particle detection, remote meter reading and people counting. The sample images for the three test applications are shown in Fig. 14. The experiments are performed 100 times in order to check the effect of changing ambient lighting conditions. With constant LED lighting, the results are good and imaging tasks are able to properly segment the objects. Table 2 shows the resource utilization of the complete design obtained by the Xilinx synthesis tool ISE [19]. For energy measurement, the average current was used to calculate the power consumption of sleep, sleep-to-wakeup, and wakeup states for one duty cycle for a 3.6 supply voltage. The instantaneous current is shown in Fig. 15. Table 3 shows the time and power required by different states for one duty cycle. The configuration time for uncompressed bit streams of 3,731,264 bits was measured to be 23.6 ms and the average current was measured to be 33.8 mA. The average current consumption at 3.6 volt resulted in an energy consumption of 2.8 mJ.

For the configuration process, the SPI bus width was set to quad-mode at a frequency of 66 MHz. The power consumption and time required by the processing and communication processes are also shown in Table 3. During processing, other components including image sensor, FLASH memory and microcontroller are also powered ON. For communication purposes, the radio transceiver was operated with a transmission throughput of 250 kbps and an output power level of 5 dBm. The micro-controller was clocked at 16 MHz in order to coordinate radio communication activities and duty cycling for the SENTIOF-CAM. The sleep current for the SENTIOF-CAM in which only the real time counter was ON, was measured to be 87 µA at 3.6 V. Table 3 includes the LED lighting energy of 48 µJ for a one frame exposure. Following this, the VSN is compared against the published systems.

A. Comparison mechanism

The direct comparison of different systems is a challenging task because of the number of parameters involved i.e.,

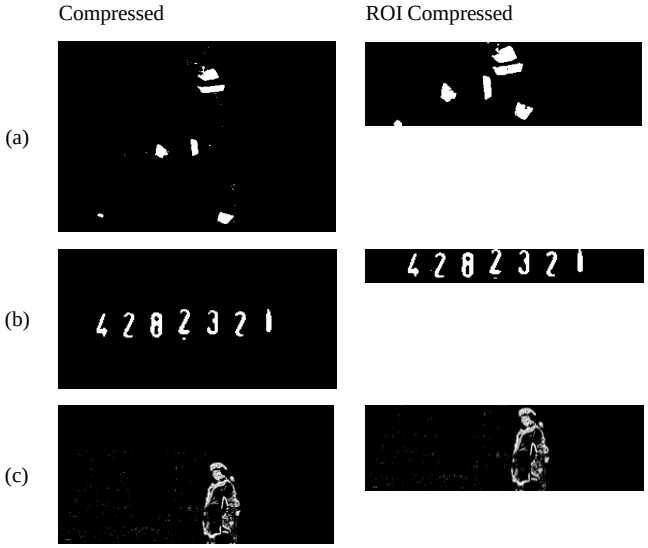


Fig. 14. Images of different applications with bi-level video coding. (a) Particle detection. (b) Remote meter reading. (c) People counting.

TABLE 2. RESOURCE UTILIZATION AND POWER CONSUMPTION OF VSN FOR DIFFERENT APPLICATIONS.* DENOTE 8K AND ** DENOTE 16K.

Resources type	**Total available resources**	**Resources used** (Particle detection+ meter reading)	**Resources used** (People counting)
Slice Registers	18224	1529	1804
Slice LUTs	9112	4460	4707
LUT Flip Flop	N.A.	5671	5804
Block RAM	64*, 32**	5*, 6**	7*, 6**

environment, lighting, vision tasks, architecture, optics, video content and resolution, etc. For an ideal comparison, these parameters should be similar. However, it is important to mention that due to limited access to other researchers' architectures, the comparison is not a fully objective process. The development of these systems for the purpose of comparison requires a significant amount of design and development efforts/costs and time.

Therefore, a suitable approach is to have a quantitative analysis based on some assumptions and provide the points of differences [30]. In the literature, a large number of systems are analyzed in [14], each with different requirements. The one to one comparison of the proposed system with the VSN system of different application requirements is a challenging task because of the unavailability of data. However, a comparison of available essential parameters i.e., power, performance and compression efficiency is given in Table 4. The comparison results in Table 4 show that the proposed system has a greater performance in terms of FPS and has an average processing power which falls close to the customized and microprocessor based solutions. The communication energy which contributes significantly to the overall energy [3] and transition states energy, are not mentioned for these systems. For detailed and fair comparison, it is necessary to select systems with the same characteristics but, it is a challenging task without the presence of standard selection criteria. The taxonomy proposed in [14] presented a mechanism for identifying a common class of systems. By using the aforementioned taxonomy we have identified a class of systems which is able to classify objects by using binary

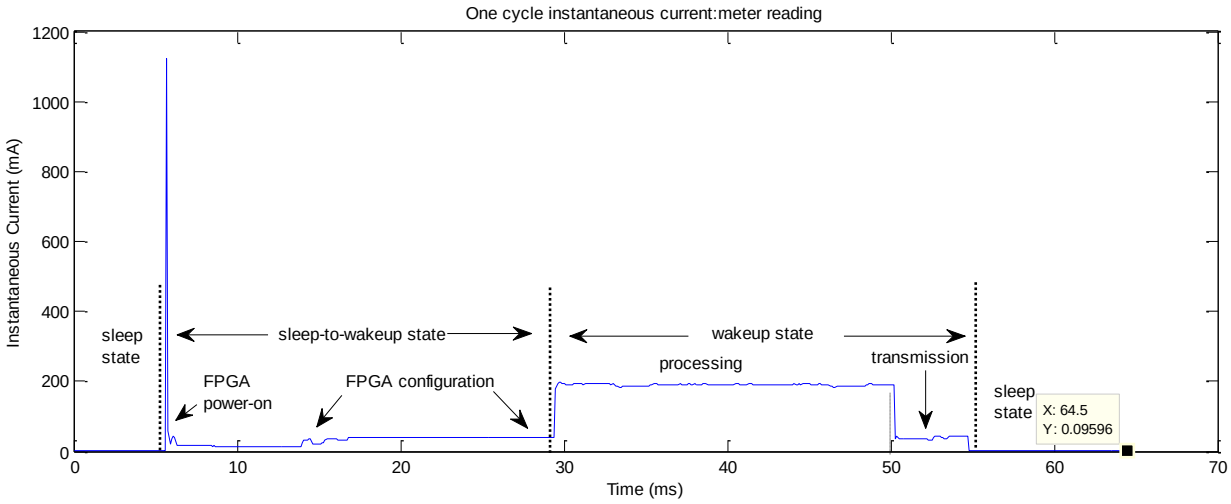


Fig. 15. VSN's instantaneous current for one duty cycle during meter reading.

TABLE 3. TIME, AVERAGE POWER AND TOTAL ENERGY FOR VSN FOR DIFFERENT APPLICATIONS.

Applications	Output data (bytes)	Config. time (ms)	Config. power (W)	Proc. time (ms)	Proc. power (W)	Comm. Time (ms)	Comm. power (W)	Total energy (mJ)
Particle detection	70	23.6	0.12	21	0.67	3.2	0.13	17.2
Meter reading	83	23.6	0.12	21	0.67	4.3	0.13	17.4
People counting	1715	23.6	0.12	16.9	0.67	63.5	0.14	22.8

TABLE 4. POWER AND PERFORMANCE PARAMETERS OF DIFFERENT VSN SYSTEMS.

VSN Systems	Processing Platforms	Resolution (W×H)	Avg. Power* (mW)	Output data (Bytes)	Bits/pixels	Frame rate (FPS)
Gasparini *et al*. [5]	FLASH FPGA	(128×64)	4.22	2048	2.0	15
Kerhet *et al*. [8]	Microcontroller + SRAM FPGA	(320×240)	500	N.A.	N.A.	15
Sanchez *et al*.[23]	DSP + FPGA	(640 × 480)	1110	N.A.	N.A.	7.5
Casares *et al*. [24]	Microprocessor	(320×240)	822	N.A.	N.A.	12
MeshEye [25]	Microprocessor	(640×480)+ (30×30)	290	900	0.02	10
CMUcam3 [13]	Microprocessor	(352×288)	500	N.A.	N.A.	14
Ferrigno *et al*. [3]	Microcontroller	(384×288)	86	6912	0.500	0.04
SENTIOF-CAM	SRAM FPGA	(640×400)	670	83	0.003	48

NOTE: * IT IS PROCESSING POWER AS COMMUNICATION POWER IS NOT SPECIFIED

data. In relation to this, three published systems including particle detection [2], remote meter reading [3] and people counting [5], having the same class as the proposed system, are identified for comparison with the proposed system.

In the particle detection system [2], the video contents, resolution and environment are similar because experiments were performed under similar conditions. For SENTIOF-CAM, the energy values were measured on real hardware in which all the components were integrated in a single prototype board. The energy values for the published particle detection system were measured for individual components with a FLASH based FPGA. The processing time of the published system did not include the configuration time which could increase the energy consumption values. In addition to this, the sleep power for the published system was taken from a datasheet, which may lead to unrealistic values. Therefore, for the processing time, the power values have been extrapolated for the published system, according to the components used in the SENTIOF-CAM. The sleep energy of the SENTIOF-CAM is considered for the published particle detection system in order to provide a fair comparison. In remote meter reading and people counting, we have performed experiments in a real environment as described in the published system. The proposed VSN and the published system have similar functionalities and the output is the same for the total system. The main difference is that in published systems, the vision tasks together with conventional compression schemes are processed locally on the VSN. In comparison to this, in the proposed system, only the pre-processing tasks together with a lightweight bi-level coding are implemented on the VSN by using a hardware reconfigurable platform whereas the control dominated tasks are moved to a server. This approach offers flexibility to use efficient algorithms for control dominated tasks i.e. detection, classification and recognition on the server. This approach will reduce design complexity on the hardware and will reduce the processing energy consumption [30]. For fair comparisons, we have performed the experiments in a real environment for 100 different samples in order to obtain average values for the output data.

TABLE 5. COMPARISON OF PROPOSED VSN WITH MEASURED ENERGY VALUES AGAINST PUBLISHED SYSTEMS FOR ENERGY CONSUMPTION, OUTPUT DATA AND PERFORMANCE.

Applications	Systems	Image size (width×height)	E_Proc (mJ)	E_Comm (mJ)	Avg. Output data (bytes)	Bits/ pixel	Max. Freq. (FPS)
Particle detection	Published [2]	(640×400)	9.6	3.52	500	0.016	48
	Measured	(640×400)	13.9	0.44	70	0.002	48
Meter reading	Published [3]	(384×288)	430.3	207.9	6912	0.500	0.04
	Published_scaled [3]	(640×400)	515.3	478.3	16000	0.500	0.03
	Measured	(640×400)	13.9	0.58	83	0.003	48
People counting	Published [5]	(128×64)	0.004	4.29	2048	2.000	15
	Published_scaled	(640×320)	0.08	27.06	51200	2.000	5
	Measured	(640×320)	11.5	8.6	1715	0.067	16

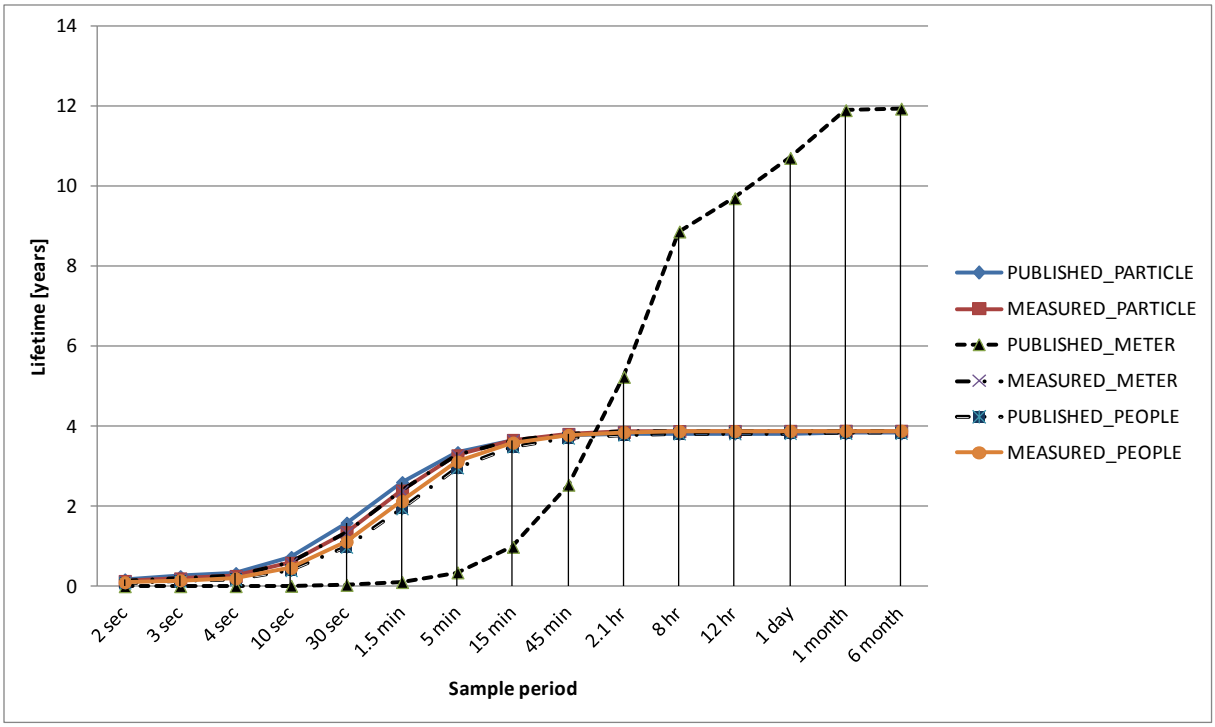


Fig. 16. VSN Lifetime over a different sample rates for published and proposed systems.

A brief discussion relating to the comparison of energy consumption, output data reduction and performance is provided in the following section.

B. Energy, output data and performance comparison

The comparison of energy consumption for the proposed VSN against the published systems is shown in Table 5. In Table 5, E_Proc shows the processing energy consumption, E_Comm shows the communication energy consumption and FPS is the frames per second. For particle detection, the published system being selected is implemented on a FLASH based FPGA [2]. The comparison in Table 5 shows that the processing energy of the SENTIOF-CAM for particle detection increases by approximately 1.4 times but the communication energy reduces by approximately 8 times. The processing energy of the FLASH based system is different from previous values because, to offer a fair comparison, the energy consumption of the image sensor AVR32 and the FLASH memory is also included, in order to have a similar setup to that for the SENTIOF-CAM. With regards to the performance parameter in Table 5, the investigated system frame rate is the same as that for the published system. This shows that the SRAM FPGA based VSN can achieve similar results to that for a FLASH based FPGA.

For remote meter reading, the proposed solution can offer up to a 69 times energy reduction as compared to the existing solution as shown in Table 5. With regards to individual components, for the proposed solution, processing energy is reduced by approximately 37 times and the communication energy is reduced by approximately 825 times. One of the reasons is that the published solution has been implemented by using a software platform. The performance of the system is increased to 48 FPS from 0.03 FPS. The processing energy in Table 5 is different to that of the previously published work [30] because of different processing platforms and coding modification for SRAM FPGA as discussed in section IV-C. In relation to the people counting application, the proposed solution offers an approximately 3 times smaller communication energy for the same resolution. In addition to this, the proposed solution offers a 3 times greater performance in terms of FPS while still having programmability as compared to the published customized based solution. The conclusion which can thus be drawn is that strategies are required to be devised for customized solutions which can efficiently handle both the processing and communication energy of the VSN.

C. Life time

The lifetime prediction is based on measured energy values of *sleep-to-wakeup*, *wakeup* and *sleep state*s. The lifetimes of the systems for particle detection, remote meter reading and people counting are given in Fig. 16. For the lifetime calculation, the active time was considered to be 100 ms and after this, the VSN is required to be switched to a sleep state. A battery with a capacity of 37.44 kJ energy was considered for the lifetime calculation. It is assumed that the battery would offer a constant performance and zero leakage. The lifetime for the published particle detection system is approximately 3.5 years, whereas, the lifetime of the proposed VSN is approximately 3.2 years with a sample period of 5 minutes.

The lifetime for a published remote meter reading VSN with an 8 bit PIC microcontroller is approximately 0.34 years (4 months), whereas, the lifetime of the proposed VSN is approximately 3.2 years with a sample period of 5 minutes. It is, however, noted that for greater sample rates such as those greater than 45 minutes, the lifetime of the published system increases because of the smaller sleep power (0.09 mW) consumption of the published solution. This shows that for high sample rates, the proposed solution offers good results, whereas for low sample rates, the published system offers good results. The lifetime of the published VSN system for the people counting application is 2.9 years, whereas, the lifetime for the FLASH based FPGA and microcontroller contribute to the proposed system is 3.11 years. In the published system, sleep energy, whereas, in the proposed system, only the microcontroller contributes to the sleep energy. It is thus concluded that a SRAM based FPGA can be effectively used for duty cycled WVSN applications. It is further concluded that a VSN with a smaller energy consumption, a generic architecture and low complexity can be achieved by using the design matrix of tasks' partitioning, bi-level video coding, low complexity background subtraction and duty cycling on existing reconfigurable platforms.

ACKNOWLEDGMENT

The authors would like to thank Aart Mulder and M. A. Yousaf for the support in the transmission and sensor modules.

VI. CONCLUSION

In this work, a low complexity, energy efficient and reconfigurable wireless vision sensor node architecture has been presented by using a SRAM FPGA. The proposed sensor node architecture has been the result of an analysis, which considers tasks' partitioning between the sensor node and server, low complexity background subtraction, bi-level video coding and duty cycling on a SRAM based FPGA. In our proposed sensor node architecture, the initial data dominated vision tasks are implemented locally on the SRAM FPGA and the control dominated complex tasks are processed on the server. For the smaller memory requirement and duty cycling scenario, a low complexity background model was developed with the assistance of image scaling techniques in order to store a scaled version of a background image in the FLASH memory. For the subtraction operation, the image is upscaled by using a suitable interpolation technique. By using a scaling factor of 8, for both the height and width, the proposed subtraction model reduced the memory requirement by a factor of 64.

This work investigates the use of duty cycling for a SRAM FPGA based VSN. The investigation based on an actual hardware implementation shows that the SRAM FPGA based sensor node can effectively utilize duty cycling for energy

conservation. The measured energy values shows that a sensor on the SRAM based FPGA can achieve a lifetime of 3.2 years for a sample period of 5 minutes, for the available 37.44 kJ energy. The conclusion thus drawn is that a SRAM FPGA based wireless vision sensor node can achieve a smaller energy consumption, a generic architecture and low complexity by using the proposed design matrix of tasks partitioning, low complexity background subtraction, bi-level video coding and duty cycling.

REFERENCES

[1] S. Hengstler and H. Aghajan, "A Smart Camera Mote Architecture for Distributed Intelligent Surveillance", *ACM SenSys Workshop on* Distributed Smart Cameras, Oct. 2006.

[2] M. Imran, K. Khursheed, N. Lawal, M. O'Nils, and N. Ahmad,"Implementation of Wireless Vision Sensor Node for Characterization of Particles in Fluids", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 22, pp. 1634 - 1643, Nov. 2012.

[3] L. Ferrigno, S. Marano, V. Paciello, and A. Pietrosanto, "Balancing computational and transmission power consumption in wireless image sensor networks," *Proc. of the IEEE Intl. Conf. on Virtual Env. Human-Computer Interfaces, and Meas. Systems*, Jul. 2005, pp. 61–66.

[4] S. Soro and W. Heinzelman, "A survey of visual sensor networks", *Advances in Multimedia*, vol. 2009, May, 2009.

[5] L. Gasparini, R. Manduchi, M. Gottardi, D. Petri, "An Ultralow-Power Wireless Camera Node: Development and Performance Analysis", *IEEE Trans. on Instr. and Meas*, vol. 60, pp. 3824 - 3832, Dec. 2011.

[6] B. Rinner , M. Quaritsch , W. Schriebl , T. Winkler and W. Wolf "The evolution from single to pervasive smart cameras", *Proc. ACM/IEEE Int. Conf. Distributed Smart Cameras*, pp.1 -10, Sep. 2008.

[7] M. Bakkali, R. Carmona-Galá, A. Rodriguez-Vazquez, "A prototype node for wireless vision sensor network applications development", *5th Intl Sym. on I/V Comm. and Mobile Network*, Sep. 2010, pp. 1-4.

[8] A. Kerhet, M. Magno, F. Leonardi, A. Boni and L. Benini "A low-power wireless video sensor node for distributed object detection", *journal of real-time image processing*, vol.2, Dec. 2007, pp 331-342.

[9] A. Kandhalu, A. Rowe, R. Rajkumar, "DSPcam: A Camera Sensor System for Surveillance Networks", *Third ACM/IEEE Intl. Conf. on Distributed Smart Cameras*, Sep. 2009, pp. 1-7.

[10] I. Dietrich and F. Dressler, "On the lifetime of wireless sensor networks,"*ACM Trans. on Sensor Net.*, vol. 5, pp. 1–38, Feb. 2009.

[11] M. Noori and M. Ardakani, "Optimal Node Distribution for Achieving a Desired Lifetime in Wireless Sensor Networks," *24th Biennial Symposium on Communications*, 2008, pp. 258-261.

[12] M. Hempstead, M. J. Lyons, D. Brooks, and G.Y. Wei, "Survey of hardware systems for wireless sensor networks". Journal of Low Power Electronics. April 2008, Vol. 4, pp.11-20.

[13] A. Rowe, A. Goode, D. Goel, I. Nourbakhsh. (2007). *CMUcam3: An Open Programmable Embedded Vision Sensor* [online]. Available: http://www.cmucam.org.

[14] M. Imran, K. Benkrid, K. Khursheed, N. Ahmad, M. A. Waheed, M. O'Nils, N. Lawal "Analysis and characterization of embedded vision systems for taxonomy formulation", *SPIE Real-Time Image and Video Processing*, USA, Feb. 2013.

[15] D. G. Bailey, *Design for Embedded Image Processing on FPGAs*, Asia, John Wiley & Sons, (2011), pp. 199-231.

[16] G. Anastasi , M. Conti , M. Di Francesco and A. Passarella "Energy conservation in wireless sensor networks: A survey", *Ad Hoc Networks*, vol. 7, pp.537 -568, May 2009.

[17] W. J. MacLean,"An Evaluation of the Suitability of FPGAs for Embedded Vision Systems". *IEEE Computer Society Conf. on Comp. Vision and Pattern Recognition*, Jun. 2005, pp. 25-25.

[18] L. Gasparini, R. Manduchi, M. Gottardi, and D. Petri, "Performance analysis of a wireless camera network node," *IEEE Instrumentation and Measurement Technology Conference*, 2010, pp. 1331-1336.

[19] Xilinx, Inc, *Spartan-6 family and Xilinx power tools tutorial, Xilinx ISE 13.1* (2010) [online]. Available: http://www.xilinx.com/.

[20] Microsemi Inc., *IGLOO Low-Power Flash FPGAs Datasheet* (2012) [online]. Available: http://www.microsemi.com/.

[21] V. Juan, O. Andres, L. Miguel, P. Jorge, T. De la Eduardo, and R. Teresa, "Using SRAM Based FPGAs for Power-Aware High Performance Wireless Sensor Networks", *Sensors*, Vol. 12, pp. 2667-2692, Feb.2012.

[22] P. Babaei, "Vehicles tracking and classification using Traffic zones in a hybrid scheme for intersection traffic management by smart cameras", *Intl. Conf. on Signal and Image Processing*, pp. 49–53, 2010.

[23] J. Sánchez, G. Benet, J. E. Simó, "Video sensor architecture for surveillance applications", *Sensors, vol.* 12, pp. 1509–1528, Feb. 2012.

[24] M. Casares, S. Velipasalar, "Adaptive Methodologies for Energy-Efficient Object Detection and Tracking With Battery-Powered Embedded Smart Cameras", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, pp. 1438 – 1452, Oct 2011.

[25] S. Hengstler, D. Prashanth, F. Sufen, H. Aghajan, "MeshEye: A Hybrid-Resolution Smart Camera Mote for Applications in Distributed Intelligent Surveillance", *6th Intl. Symposium on Information Processing in Sensor Networks*, Apr. 2007, pp.360-369.

[26] M. Paavola, K. Leiviska, "Wireless Sensor Networks in Industrial Automation", J. *S. Blanes (ed.), Factory Automation, Ch 10, InTech*, pp. 201-220, 2010.

[27] P. Cheng, Y. Yang, B. Oelmann, "Design and Implementation of a Stator-Free RPM Sensor Prototype Based on MEMS Accelerometers", *IEEE Transactions on Instrumentation and Measurement*, vol.61, no.3, pp.775-785, Mar. 2012.

[28] K. Shahzad, P. Cheng, and B. Oelmann, "SENTIOF: An FPGA-Based High-Performance and Low-Power Wireless Embedded Platform" *International Conference on Wireless Sensor Networks*, 8-11 Sep. 2013.

[29] Aptina Imaging Corporation., *MT9V011, CMOS Image Sensor* (2009) [online]. Available: http:// www.aptina.com/.

[30] M. Imran, N. Ahmad, K. Khursheed, M. A. Waheed, N. Lawal, M. O'Nils, "Implementation of Wireless Vision Sensor Node With a Lightweight Bi-Level Video Coding", *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol.PP, pp.1,12, May. 2013.

[31] H. Norell, N. Lawal, M. O'Nils, "Automatic Generation of Spatial and Temporal Memory Architectures for Embedded Video Processing Systems". *Eurasip Journal on Embedded Systems*, 2007.

[32] Winbond Electronics Corporation, *W25Q64BV, Winbond Electronics*, (2013) [Online]. Available: http:// www.winbond.com/.

[33] E. Aho, J. Vanne, T.D. Hämäläinen and K. Kuusilinna, "Configurable Implementation of Parallel Memory Based Real-time Video Downscaler", *Microprocess.Microsyst*, vol. 31, pp. 283–292, Aug. 2007.

[34] C.-H. Kim, S.-M. Seong, J.-A. Lee, and L.-S. Kim, "Winscale: an image-scaling algorithm using an area pixel model", *IEEE Transactions on Circuits and Systems for Video Technology* 13 (6) pp.549–553, June 2003.

[35] M. Imran, N. Ahmad, K. Khursheed, M. O'Nils and N. Lawal, "Low Complexity Background Subtraction for Wireless Vision Sensor Node", *16TH Euromicro Conference on Digital System Design*, Spain, Sept. 2013.

[36] J. Wunschmann, S. Zanker, C. Gunter and A. Rothermel, "Reduction of computational cost for high quality video scaling", *IEEE Trans. Consumer Electron*., vol. 56, pp. 2584-2591, 2010.

[37] Adobe Developers Association, *TIFF, Revision 6.0* (1992), [online]. Available: http://www.itu.int/itudoc/itu-t/com16/tiff-fx/docs/tiff6.pdf

[38] Agilent Technologies., *Agilent 34410A*, (2013) [Online]. Available: http://www.home.agilent.com/.

[39] D. Puccinelli and M. Haenggi "Wireless sensor networks: Applications and challenges of ubiquitous sensing", *IEEE Circuits Syst. Mag*., vol. 5, no. 3, pp.19 -31 2005.

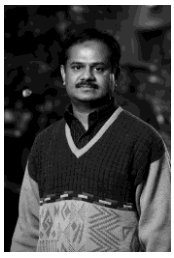
[40] Atmel Corporation, *AT32UC3B0256, AVR32* (2010) [online]. Available: http://www.atmel.com/.

[41] Texas Instruments Inc, *CC2520 transceiver* (2007) [online]. Available: http://www.ti.com/.

[42] Digilent Inc, *RS232 Reference Project, Adept 2.10.2* (2011) [online]. Available: http://www.digilentinc.com/.

Muhammad Imran received B.S. degree in computer engineering from COMSATS Institute of Information Technology, Abbottabad, Pakistan, in 2006, the Masters degree in electrical engineering (System-on-Chip) from Linköping University, Sweden, in 2009, the Licentiate and PhD degrees in electronics design from Mid Sweden University, Sweden, in 2011 and 2013 respectively. Imran is currently working as Research Fellow in Mid Sweden University, Sweden. His research interests include design, development and implementation methodologies for embedded video processing systems both on software and hardware and investigation of energy efficient and reconfigurable architectures for wireless smart cameras.

Khurram Shahzad received the M.Sc. degree in electrical engineering with specialization in System-on-chip from Linköping University, Linköping, Sweden, in 2009. He is currently pursuing the Ph.D. degree in electrical engineering at Mid Sweden University, Sundsvall, Sweden. His current research interests include digital systems, sensor-based systems for industrial applications, and embedded system design.



Naeem Ahmad received M.Sc. degree in electronics from Quaid-e-Azam University Pakistan, Masters degree in electrical engineering (System-on-Chip) from Linköping University Sweden, and Licentiate of Electronics and PhD degrees from Mid Sweden University Sweden. Currently, he is working in Nuclear Institute for Agriculture and Biology, Pakistan. His main research interest includes modeling and optimization of visual sensor networks and design of vision based surveillance system.

Mattias O'Nils received his B.Sc. electrical engineering degree from Mid Sweden University in 1993 and his Licentiate/PhD degrees in electronics system design from The Royal Institute of Technology in Stockholm, Sweden, in 1996 and 1999 respectively. He is Professor/prefect of the department and leads a research group in embedded systems design at Mid Sweden University. His research interest include hardware/software partitioning, methods for performance optimization by migration of functionality to hardware, specification, design and verification of hardware/software interfaces, low-power optimization of FSM implementation, compiler technology memory optimization and synthesis for real-time video systems.

Najeem Lawal received his B.Sc. degree in Electrical Engineering from the University of Lagos, Nigeria in 1998. He received his M.Sc. and PhD. in electronics system design from the Mid Sweden University, Sundsvall, Sweden in 2004 and 2009 respectively. His research involves machine vision where a combination of state-of-the-art development in software and hardware are employed in implementing the systems. In particular, Najeem focuses on the power consumption of vision system algorithms and works on how it can be optimized through algorithm optimization and implementation platform exploration.

Bengt Oelmann received the Doctor of Technology degree in electronics from the Royal Institute of Technology, Stockholm, Sweden, in 2000. He is currently a Full Professor in electronics system design with Mid Sweden University, Sundsvall, Sweden. From 1996 to 1998, he was a Senior Application-Specific Integrated Circuit (ASIC) Designer with Nordic very-large-scale integration (VLSI), Norway. He has published more than 70 reviewed papers in the fields of low-power integrated circuit design, pixelarray sensor electronics, video coding, and industrial electronics.