Journal of Theoretical and Applied Information Technology
15th February 2022. Vol.100. No 3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

RECONFIGURABLE SOC ARCHITECTURE FOR WIRELESS VIDEO SURVEILLANCE

*[1]Dr. P. LATHA, [2]Dr. M. A. BHAGYAVENI, [3]Dr. S. R. PREETHI, [4]S. SIMCY

[1]Associate Professor, St. Joseph's College of Engineering, Department of ECE, Chennai, INDIA
[2] Professor, College of Engineering, Guindy, Department of ECE, Anna University, Chennai, INDIA
[3]Assistant Professor, Valliammai Engineering College, Department of ECE, Chennai, INDIA
[4]Assistant Professor, Vins Christian college of engineering, Department of ECE, Chennai, INDIA

E-mail: [1]lathagct97@gmail.com, [2]bhagya@annauniv.edu, [3]srpreethi31090@gmail.com, [4]simcybinu@gmail.com

ABSTRACT

The advancements in SoC (System on Chip) technology make the design engineers to implement the function of wireless sensor network node in a single chip. Reconfigurable System on Chip (RSoC) plays a vital role in video surveillance applications, where detection of surveillance objects is generally achieved by subtracting estimated background from the raw video. So, this paper proposes a reconfigurable SoC architecture for wireless video sensor network node capable of extracting a moving object in video surveillance system using background subtraction algorithm. The proposed video surveillance node aims in performing real-time moving object detection using FPGA (Field Programmable Gate Array) hardware on high resolution video sequences with a frame size of 4608 × 3456. The proposed system processes 10fps (frames per second) and, therefore, the processing capability is 159.3 Mps (Mega pixels per second) of very high system performance. The background subtraction algorithm is implemented in VIRTEX-5 FPGA kit. The RS232 serial port cable is used for establishing the connection between hardware and PC. The FPGA resource utilization is linearly increased with respect to pixel width of the frame. The area taken and the speed of the algorithm are also evaluated. The proposed hardware implementation is compared with the software implementation running on a 2.10 GHz Intel i3-2310M processor. The hardware implementation is 26.36 % faster than software implementation for complex video inputs.

Keywords: *Reconfigurable SoC, Wireless sensor network (WSN), Surveillance, object detection, Background subtraction, Video processing*

1. INTRODUCTION

Now-a-days Wireless sensor networks (WSN) are used broadly for different security and surveillance applications. Video surveillance has turned into a vital part of industrial security and operations monitoring. Video Surveillance in WSN gives an extremely flexible method for monitoring outdoor regions, for example hospital and secretariat campuses, parking areas in highly secured industrial plants like atomic power plant and building sites. In video surveillance applications, detection of surveillance objects is generally achieved by subtracting estimated background from the raw video. Moving object detection is an important problem in video analysis. It is necessary for many video processing applications such as video surveillance applications, vehicles detection for efficient video compression and smart tracking of moving objects. Many high-level computer vision tasks like navigation of robot, collision avoidance, video surveillance and path planning need tracking of the moving objects in the surrounding environment in real time.

1.1 Overview of Video Surveillance System

Smart surveillance, intelligent video surveillance, video analytics, intelligent video and intelligent analytics are typical names used to describe the concept of applying automated signal analysis and pattern recognition to video cameras and sensors, with the goal of automatic extraction of "usable information" from video and sensor streams. Smart Surveillance helps optimize security by integrating hardware, software and services within an organization, thereby enabling the convergence of physical and IT security. The most important challenge in such systems is to manage the huge information flow in the network nodes. The systems with various non-visual and visual

sensors acquire and, should process in real-time, large amounts of data. The most straightforward solution is to provide the sensor nodes with a certain level of autonomous intelligence so that the captured data can be locally processed and analyzed, while only the observation of significant importance or unknown situations would be reported to the control room.

1.1.1 The modules of a surveillance system

A surveillance system consists of three modules: source, functional and sink modules [1] as shown in Figure 1.

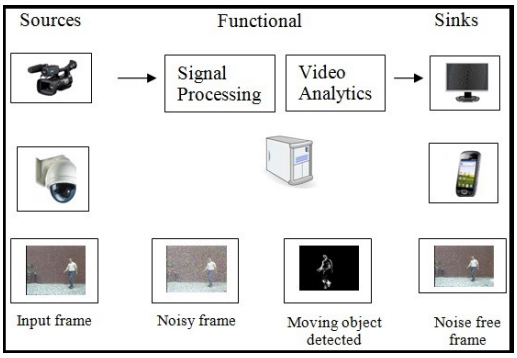

Figure 1: Modules in a video surveillance system

Source Modules: The source module of the surveillance systems can be equipped with sensors belonging to different domains to capture the input from the multi-sensory nature of the events. The sensors that can be connected in a source module are image sensor, audio sensors, ID sensors, (e.g. biometric sensors like fingerprint, RFIDs, bar codes, etc.) and context sensors (e.g. temperature, PIR, etc.). Since the nature of the sensed events is analog, the source module is an analog unit and it is followed by an analog-to-digital converter, in order to perform digital signal processing.

Functional Modules: The functional modules process the input and provide an output on the same or on a different domain. They can be divided into two parts: The first part is the Signal Processing Modules used to modify or transform the properties and the quality of the input. This module consists of analog-to-digital converters, signal encoders and decoders, streamers and sensor pre-processing unit. In case of video surveillance system, the pre-processing unit performs quality enhancement (SNR-Signal to noise ratio), pixel-wise operations (e.g. filtering, morphology, etc.) and region-of-interest operations (resizing, cropping, etc.): the pre-processing precedes analysis of higher complexity (e.g. video analytics), and it can be directly connected to the camera. The second part of the functional module is Purposive Modules devoted to extract information from video. This module consists of a software called 'video analytics' based on computer vision and pattern recognition techniques, ranges from the most basic motion detection, up to the new generation of surveillance systems for event detection, behavioral analysis, situation assessment and automatic attention focusing.

Sink Modules: The sink modules are the display unit used for displaying the processed video to the end user.

1.1.2 Processing unit in video surveillance system

Processor selection plays a larger role in the success of developers in meeting the design requirements of digital video surveillance products. Options include more powerful microcontrollers, Digital Still Camera (DSC) processors, Application-Specific Integrated circuits (ASIC), Digital Signal Processors (DSP) and FPGA. FPGAs are presented as a very suitable solution for high performance WSN Video Surveillance applications. With its powerful parallel processing, theoretically the FPGA would excel at high-end video surveillance because of the complex and intensive video signaling required.

As chip fabrication techniques continue to advance and become more refined, the concept of SoC will further evolve and grow in popularity. With system components moved from on-board to on-chip, communication time and bandwidth are greatly improved, raising the question of exactly what type of hardware to include on SoCs. Reconfigurable hardware shows great potential for SoC use, providing hardware speeds, while maintaining a level of flexibility not available with traditional custom circuitry. This flexibility is the key to allowing both hardware reuse and post-fabrication modification.

The core of a reconfigurable architecture is a set of hardware resources, including logic and routing, whose function is controlled by on-chip configuration SRAM. Programming the SRAM, either at the start of an application or during execution, allows the hardware functionality to be configured and reconfigured, permitting reconfigurable systems to implement different algorithms and applications on the same hardware. This reusability makes reconfigurable hardware a

prime candidate as a subsystem for SoCs. Rather than using separate custom circuits to accelerate each potential application, a single reconfigurable architecture can be used. This reconfigurable logic can implement circuits from each application in hardware as needed. FPGAs are a widely available form of reconfigurable hardware. Reconfigurable architectures can be made more efficient if the algorithm types are known in advance.

The growth of wireless sensor networks was initially inspired by military applications such as battlefield surveillance. These extraordinary networks are proposed to be self-governing and reconfigurable. When the wireless sensor node is designed with a reconfigurable FPGA, the node has enormous advantages like the functions of the WSN node can be modified, extended or redesigned after shipping the product to the application field.

1.2 RSoC Wireless Video Sensor Node

Reconfigurable System on Chip (RSoC) plays a vital role in video surveillance applications. While designing a sensor node as RSoC, Designers try to integrate CMOS, MEMS and RF communication technologies to create extremely low power and miniaturized sensor nodes that still provide certain sensing, computation, reconfiguration and communication capabilities. Among the sensor node hardware platforms, the Berkeley motes [2] are an example for open source software development and commercial availability. These motes have gained wide popularity among the sensor network research community. Reconfigurable hardware can improve the processing systems performance. A reconfigurable device allows not only performance improvement but also remote hardware reconfiguration of the WSN node. Hardware reconfiguration features improve node flexibility and computation performance, and opens the possibility of remote hardware reconfiguration, which can be very useful to tune node performance with new actualizations, to debug on-line commissioning of the WSN and make the node smarter.

In this paper, a reconfigurable SoC for a wireless video sensor node is proposed to detect a moving object in surveillance application. Background Subtraction Algorithm is used for moving object detection. The proposed reconfigurable SoC-based wireless video sensor node processes high resolution video frames of 4608 × 3456 and the processing capability is 159.3 Mps.

The rest of this paper is organized as follows: The related work is discussed in section 2. In section 3, an overview of the proposed wireless video sensor node architecture is discussed. Analysis of the performance of the proposed algorithm and result are discussed in section 4. Concluding remarks and future works are presented in the last section.

2. RELATED WORK

Work related to Reconfigurable SOC architecture for wireless video surveillance can be found in the literature. Table 1 compares the related work in RSoC WSN video surveillance node with moving object detection techniques. The table indicates the kind of algorithm model that is used and the use of different FPGA devices with its speed in the first and second sections. The third section reports the video signal resolution in terms of pixels and frame rate (FPS). The fourth section of the table shows the requirements in terms of resources utilized: lookup-tables (LUTs), Flip Flops, Registers, Slices, Arithmetic Logic gates, Bonded IOBs (Input Output Blocks) and Block RAMs (BRAM). The table reports the applications where the algorithms are employed in the last section.

An implementation of the Multiple of Gaussian (MoG) algorithm is presented in [3]. Three different methods like, a unimodal grey-scale background model, a unimodal colour (RGB colour space) background model and a bi-modal grayscale background model were described. All methods were implemented on a Xilinx Virtex II device with 4 external ZBT RAM memory banks and with a maximal frequency of about 65 MHz.

Table 1: Comparison Of Related Work In Rsoc WSN Video Surveillance Node With Moving Object Detection Techniques

NA: Not Available

Work and Year	Model	Video signal Resolution	FPGA Device / Speed	Resource Utilization	Applications
Appiah & Hunter [3], 2005	Adaptive background models	640X480 pixels at up to 210 fps	Xilinx Virtex II XC2v6000 57MHz to 65MHz	Flip Flops: 1,766 LUTs: 3,34 Block RAMs: 57 bonded IOBs: 366 Slices: 2,124	The extraction of accurate background models under variable lighting conditions in real-time. Automated visual surveillance systems.
Hongtu Jiang et al [4], 2005	Statistical background modeling.	1024×1024 pixels 38 fps	Xilinx Virtex2 1000 40MHz	NA	video segmentation for intelligent video surveillance
Jozias Oliveira et al [5], 2006	Background subtraction	240 x 120 pixels 30 fps	FPGA 40 MHz	NA	Moving Region Segmentation
Gorgon et al [6], 2007	Background generation algorithm & Sum of Absolute Difference (SAD) algorithm	576 ×768 pixels 25fps	XC2V6000 Virtex II	Slices: 4,902 IOBs: 385 Block RAMs: 6 Arithmetic-Logic Gates: 549,190	Video detection of vehicles.
N. T. Vu Dang et al [7], 2019	Background Subtraction With Gaussian Mixture Model (GMM)	NA	Matlab and Modelsim Simulation	NA	Vehicle Detection
P. G. Bhat et al [8], 2021.	An adaptive GMM-based background subtraction method and a three-level multi-motion modeled particle filter framework	NA	Framework	NA	Real-Time Traffic Surveillance System
Abutaleb et al [9], 2009	Multimodal Σ–Δ Background Estimation	768 x 576 pixels 1198 fps	Xilinx Spartan-II 2s200fg456 265.3 MHz	347 CLB slices	To extract the moving objects in complex scenes at a high frame rate.
YingLi Tian et al [10], 2012	Gaussian mixture Model	NA	Simulation	NA	A new framework to robustly and efficiently detect abandoned and removed objects in complex environments for real-time video

Work and Year	Model	Video signal Resolution	FPGA Device / Speed	Resource Utilization	Applications
					surveillance.
S. Li et al [11], 2021.	Embedded BackGround Subtraction (EBGS) scheme and Additive Increase Multiplicative Decrease (AIMD) filtering and random update to realize the trade-off between accuracy and efficiency.	480 × 720 20 FPS	A single-board computers (SBC): a 900MHz quad-core CPU, and 1GB RAM and implemented in C++ with OpenCV libraries.	NA	Moving Object Detection on General-Purpose Embedded Devices
M. Benetti et al [12], 2018.	Adaptive Background Subtraction and Image Segmentation	104×104 pixels 30 fps	SPARTAN 6 FPGA kit 48MHs	LUTs 1595 REGISTERS 383 BRAM(bits 16000	Low-Power Vision System for Unusual Event Detection
L. Li et al [13], 2021.	Non-convex sparsity based Background Subtraction model using Generalized Shrinkage Thresholding Operator (GSTO)	NA	Simulation	NA	Intelligent Video Surveillance
T. Nivetha et al [14], 2020.	Background subtraction method along with Sobel's edge detection algorithm to trace the outline shape of the detected moving object.	8×8 pixels	Spartan 3 FPGA kit using MATLAB and Xilinx Platform Studio (XPS) tool	NA	Soft real time application for Video-Object Detection
N. S. Sakpal et al [15], 2018	Selective Background Subtraction technique	NA	Simulation	NA	To subtract unnecessary background from the foreground and background scene.

The authors in [4] implemented the MoG algorithm (in RGB colour space) on a Virtex II 1000 FPGA. They described a module which performed all MoG stages; however the research was carried out in a simulation phase. Another background subtraction system using the Horpraset method (unimodal) in RGB colour space was described in [5]. The design was implemented on two FPGA devices and operated in real-time at resolution 240 x 120 and 30 fps with processing capability of 0.864 Mps.

An FPGA based road traffic detector using unimodal, grey-scale background generation (pixel averaging method) was described in [6]. The application operated in real-time at resolution 720 x 576 and 25 fps with 10.37 Mps processing capability. A road safety systems based on vehicles detection using Background Subtraction algorithm

with Gaussian Mixture Model (GMM) was proposed and implemented on FPGA using parallel architecture [7]. A fully automatic and real-time tracking algorithm to track multiple vehicles in a video [8] specifically tries to address the challenges of occlusion and fast-motion in traffic surveillance. The algorithm begins with automatic detection of the moving targets by an adaptive GMM-based background subtraction method.

A multimodal moving object detection method using a grey-scale, multimodal sigma-delta approach was presented in [9]. The system was implemented on a Xilinx Spartan II FPGA device and operated at 265 MHz, although the communication with external RAM was not considered. A hierarchical filtered motion (HFM) method was proposed to extract motion information and reduce the distracting motions that are caused by the background moving objects [10]. An embedded background subtraction (EBGS) scheme [11], a full-process computing optimization, which employs contracted codebook, Additive Increase Multiplicative Decrease (AIMD) filtering and random update to realize the trade-off between accuracy and efficiency was proposed. The EBGS scheme was evaluated on single-board computers (SBC): a 900MHz quad-core CPU and 1GB RAM and implemented in C++ with OpenCV libraries. They processed the image with a resolution of 480 × 720 and the processing speed was 20 FPS.

A smart ultra-low power vision system targeted to video surveillance applications [12] was descried. The sensor embeds a low-level image processing technique that autonomously detects unusual events occurring in the scene, relying on adaptive background subtraction. The resulting binary image is then directly segmented by an FPGA, which triggers the higher layer of processing, transferring only aggregate feature information. The on-board processing relieves the rest of the vision system from expensive computation. The 104 × 104 pixels vision chip consumes 80 μW at 30 frames/s. An intelligent video surveillance system based on Generalized Shrinkage Thresholding Operator (GSTO) [13], a new non-convex sparsity model was designed to preserve more information while detecting the surveillance objects by subtracting estimated background from the raw video.

Hardware based moving object detection using background subtraction method for soft real time application [14] was implemented using MATLAB tool and Xilinx ISE in Spartan 3 FPGA kit. RS 232 serial port cable was used for establishing the connection between hardware and PC. Visual basics software was used to view the output in PC. Selective background subtraction is the major problem associated with background subtraction technique. For foreground detection, background modeling is used in many different applications to subtract the background and detect foreground object in the image. N. S. Sakpal and M. Sabnis present selective background subtraction technique to subtract unnecessary background from the foreground and background scene [15].

Two-level architecture for the surveillance application in WSN was proposed in [16]. The first-level node was a microcontroller-based node interfaced with an array of basic sensors that detected a change in the environment variables. The second-level node was an FPGA-based node interfaced with camera for capturing the intruder after receiving alert signal from the first-level node. Three sensors were used for detecting the presence of any intruder in surveillance area. The first sensor was a proximity sensor that detected a change in the magnetic field in the vicinity, the second sensor was an ultrasonic sensor used to detect an intruder in its line of sight, and the third sensor was a vibration sensor used for detecting footsteps, movement of vehicles and other similar disturbances. The microcontroller-based node continuously monitored the output signals from all three sensors. Upon exceeding the threshold limit of each of these three sensors, the microcontroller sent an alert signal through wireless interface to the FPGA-based node that activated the camera for capturing the intruder.

A new reconfigurable WSN node [17] with image enhancement technique for video sequence in surveillance application was proposed. The main aim of this reconfigurable node was to detect the presence of an intruder in the environment and to determine the number of intruders based on threshold computation. The proposed node in the application was composed of a high performance FPGA that could be dynamically reconfigured for different applications, interfaced with the camera. This camera supported 1280 x 720 resolutions for video broadcasting. The video was captured at a frame rate of 30 frames per second and therefore the processing capability of the node was 27.65 Mps. This reconfigurable WSN node used Sum of Absolute Difference Algorithm (SAD) for intruder detection. The algorithm for

Histogram equalization was implemented in the proposed node for enhancing the clarity of the captured video frames.

A reconfigurable SoC architecture for ship intrusion detection interfaced with 3-axis digital accelerometer sensors [18] that could be deployed onto the sea surface was proposed to detect unauthorized ships. The proposed three-level detection system was developed in order to find effectively the intruding ship on the sea surface with the help of a 3-axis accelerometer sensor. Compared to the existing methods, the proposed system was cost effective and easy to deploy as it occupied less area and was the first reconfigurable SoC-based ship intrusion detection system.

In this paper, a reconfigurable SoC-based wireless video sensor node is proposed to detect a moving object in surveillance system using background subtraction algorithm. The next section discusses in detail the proposed system.

3. OVERVIEW OF THE PROPOSED WSN VIDEO SURVEILLANCE RSoC

The proposed system architecture for WSN video surveillance RSoC is shown in Figure 2. The functional modules implemented in this architecture are as follows

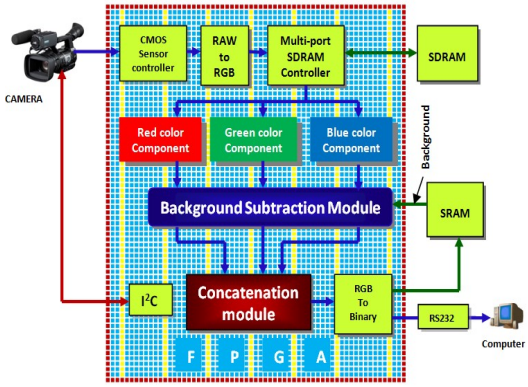

Figure 2: Proposed system architecture

3.1 CMOS Image Sensor

An image sensor is a device that converts an optical image into an electric signal. In digital cameras and other imaging devices, an image sensor plays a very important role in converting an optical image into an electrical signal. A typical CMOS is an integrated circuit with an array of pixel sensors. In CMOS sensors, each pixel sensor consists of its light sensor, an amplifier and a pixel select switch. Image resolution and frame rate are the two important parameters for selecting the image sensor. Higher image resolution can provide much more detailed information regarding the objects. In this work, 16.2 Megapixel Exmor R CMOS Sensor of model DSC-WX80/B capable of supporting 4608 x 3456 resolutions for video broadcasting was used to capture video. Videos were captured in frame rate of 10 frames per second.

3.2 CMOS Sensor Controller

It controls all the operations performed by CMOS sensors. It acts as an interface between image sensor and RAW to RGB convertor.

3.3 RAW to RGB Conversion

Raw image file contains minimally processed data from the image sensor. Raw image files are sometimes called digital negatives, that is, the negative is not directly usable as an image, but has all of the information needed to create an image. The process of converting a raw image file into a viewable format is sometimes called developing a raw image. CMOS image sensors include the color filters of an RGB Bayer array, which lets the sensor detect colors. Most consumer grade image displaying devices require an RGB image data format with red, green, and blue in each pixel's data.

3.4 SDRAM

SDRAM is the conventional Single-Data-Rate (SDR) SDRAM memory interface, which allows the storage of a large amount of data. This data can be accessed in bursts at 133MHz allowing the FPGA to process that data in real time, or to create a storage element such as a large FIFO. The SDRAM is organized as 32-bit wide memory. Its interface always runs at 133MHz. Therefore, once a row is opened, data can be written in a burst of consecutive clock cycles at a data rate of 532Mbytes/sec. RGB frames are stored in SDRAM.

3.5 SDRAM Controller

The SDRAM controller, located between the SDRAM and the bus master, reduces the user's effort to deal with the SDRAM command interface by providing a simple generic system interface to the bus master. It consists of three modules: the main control module, the signal generation module

and the data path module. The main control module, containing two state machines and a counter, is the primary module of the design which generates proper outputs according to the system interface control signals. The signal generation module generates the address and command signals required for SDRAM. The data path module performs the data latching and dispatching of the data between the bus master and SDRAM.

3.6 SRAM

The background image is stored in the external SRAM memory.

3.7 Moving Object Detection Algorithms

3.7.1 Background subtraction [19-21]

Moving object detection provides a classification of the pixels in the video sequence into either foreground (moving objects) or background. Background removal which is referred to as background subtraction is a common approach used to achieve for classifying the video sequence, where each video frame is compared against a reference or background model, pixels that deviate significantly from the background are considered to be moving objects. Background subtraction is the process of comparing an image frame to the background model to figure out if individual pixels are part of the background or the foreground. This process is also referred to as foreground detection. Background subtraction is a computing process of extracting foreground objects in a particular scene. A foreground object can be defined as an object which helps in reducing the amount of data to be processed and in providing important information to the task under consideration. In general, the foreground object is a moving object in a scene. There are many challenges in devising a good background subtraction algorithm. First and foremost, it must be more robust against changes in illumination. The next challenge is that it should avoid detecting non-stationary background objects and shadows made by moving objects. It should also deliver a good foreground detection rate and the processing time for background subtraction to be real-time. The following algorithms are also used for moving object detection.

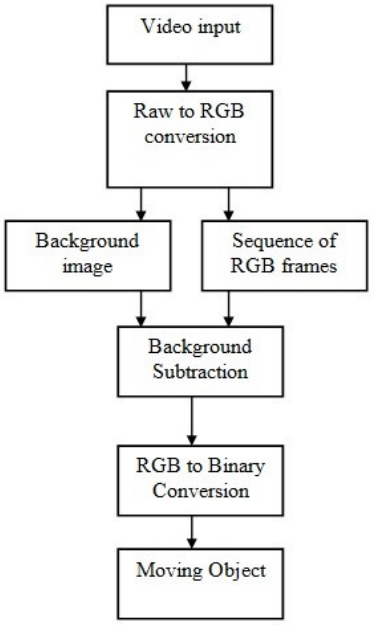

Figure 3: Block diagram of moving object detection using background subtraction technique

The block diagram for the background subtraction algorithm is given in Figure 3. At first, the raw video input is converted into RGB frames. In FPGA, the colour images need to be processed with individual colour component separately in order to reduce the memory usage. Background image is taken from SRAM and it is subtracted from the input frames one by one by using background subtraction algorithm. Finally, the moving object is identified from the image. Mathematically, the background subtraction algorithm is given by equation (1), where $f(x, y, t)$ is an image taken at time t and $B(x, y)$ is the reference image (or background).

$$d(x,y,t) = f(x,y,t) - B(x,y) \qquad (1)$$

3.7.2 Optical flow algorithm [22, 23]

Optical flow is defined as the apparent motion of brightness pattern in an image sequence. It can be considered as an approximation of motion field of each pixel in the image. Usually, optical flow estimation is carried out using spatio-temporal gradient algorithms. These algorithms work by determining the spatial and temporal bright changes (gradients) of the image gray pattern, and estimating the optical flow from these parameters. This method is based on the fact that object

movement produces changes in brightness. Optical flow methods make use of the flow vectors of moving objects over time to detect moving regions in an image. They can detect motion in video sequences even from a moving camera; however, most of the optical flow methods are computationally complex and cannot be used in real time without specialized hardware. Since it is necessary to store more than one image, the optical flow algorithm requires high memory resources.

3.7.3 Temporal difference algorithm

Temporal Difference Algorithm [24-26] attempts to detect moving regions by making use of the pixel-by-pixel difference of consecutive frames (two or three) in a video sequence. This method is highly adaptive to dynamic scene changes however; it generally fails in detecting whole relevant pixels of some types of moving objects. Also this method fails to detect stopped objects in the scene. Additional methods have to be adopted in order to detect stopped objects. Temporal difference algorithm subtracts each frame from the next frame. It reduces image pixel resolution from 10 bit data per pixel to 5 bit data per pixel. In order to store the reference image in the chip memory, the bit resolution of the image needs to be reduced. The major drawback of using the temporal difference algorithm is the difficulty of detecting the object shape.

3.8 RGB to Binary Conversion

An RGB image is converted into a binary image using equation (2). If the pixel in the subtracted frame, d(x,y,t) is greater than the predetermined threshold value T, the corresponding pixel is replaced by '1'. Otherwise, the pixels are replaced by '0'. In the dynamic image analysis, all pixels in the motion image *Bin*(*x, y, t*) with value of 1 (white) are considered as moving objects in the scene. The level of 0 (black) indicates a pixel belonging to the background.

$$Bin(x,y,t)=\begin{cases}1 & if \quad |d(x,y,t)|>T \\ 0 & Otherwise\end{cases} \qquad (2)$$

3.9 RS232 Interface

The RS-232 interface is the Electronic Industries Association standard for the interchange of serial binary data between two devices. It is commonly used in computer serial ports. It was chosen because data traffic is only related to object position and, therefore, the performance is not affected. The binary frames have to be sent to the PC through RS-232 interface.

3.10 I²C Interface

Inter-Integrated Circuit (I²C) is a two-wire serial bus typically used to connect the video camera with FPGA. It is actually a bus slave that transfers data at different speed. In earlier days, I²C bus data rates are the 100 kbit/s in standard mode and the 10 kbit/s in low-speed mode. Now-a-days I²C can be used to connect more nodes and run at data rates of the 400 kbit/s in Fast mode and 3.4 Mbit/s in High Speed mode. These speeds are more widely used in FPGA-based sensor nodes.

4. TEST AND RESULTS

This section discusses the results obtained from the FPGA implementation regarding the resources used and the processing speed reached. The results are compared with the execution of the background subtraction algorithm, implemented in MATLAB, on an Intel Core i3-2310M CPU (2.10 GHz and 4 GB of RAM memory). The design is implemented in VHDL and synthesized for a Xilinx Vertex-5 FPGA using Xilinx ISE Design suit 12.2. The reconfigurable device XC5VLX110T is 110 Trillion Logic Cells Vertex-5 FPGA. The development board consists of On-board 32-bit ZBT synchronous SRAM and 1GB Compact Flash card. The board XUPV5-LX110T has Two Flash PROMs (32 Mbyte each) and Programmable system clock generator. Figure 4 shows the original video frame and background environment for testing the module.

4(a) Input frame *4(b) background*

Figure 4: Original video frame

The sequence of 9 input frames are arranged as 3x3 matrix window which represent the different movement of the object (TOY) is shown in Figure 5.

Journal of Theoretical and Applied Information Technology
15th February 2022. Vol.100. No 3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 5: A sequence of input frames

The sequence of output frames obtained from VHDL simulation is shown in Figure 6. For color image processing, three modules can be employed to process red–green–blue (RGB) components for each pixel directly. After processing three components separately, they can be combined together and the resultant RGB frames are shown in Figure 6.

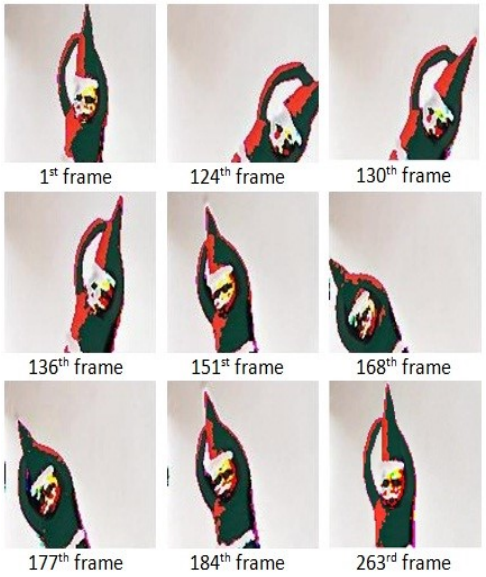

Figure 6: A sequence of RGB output frames

The RGB frames are converted into binary to extract the moving objects clearly and the sequence of binary frames is shown in Figure 7.

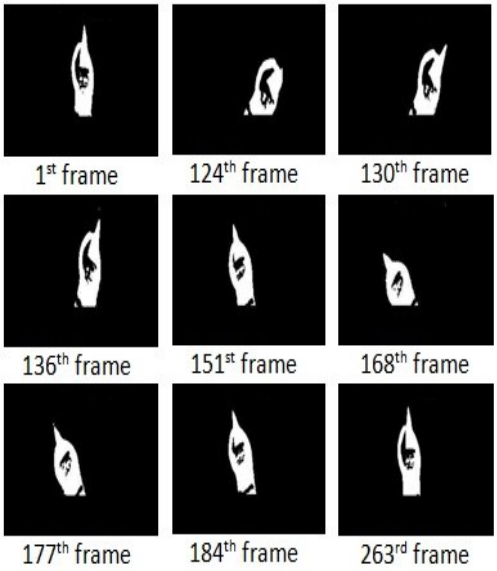

Figure 7: A sequence of binary frames converted from RGB format

Table 2 summarizes the hardware implementation of the proposed module in FPGA of different device model from Xilinx vendor. This table presents the number of utilized resources against the number of available resources, as well as the total memory usage of the proposed reconfigurable SoC-based object detection module for each device. The proposed design unit used 16 flip flops, 8 slices of 69120 available slices and 18 LUTs in XC5VLX110T Virtex 5 FPGA. The implemented module used 8 RAMs and 6.56% of the available bonded IOBs.

The hardware implementation of the proposed surveillance system in Virtex 5 FPGA is compared with other FPGA model in Table.1. The area utilization is the same for all FPGA models. Virtex 4 and Virtex 5 FPGAs take 22 seconds of total real time to synthesis completion. Moreover, the maximum output required time in Vertex 5 implementation is increased by 54.52% in Spartan 3, 14.66% in Spartan 6 and 22.79% in Vertex 4 implementations and is reduced by 76.16% in Vertex 6 FPGA implementation as shown in Table.1 and the graph is plotted in Figure 8. The total memory usage is higher in Vertex 5 implementation of the proposed design unit when compared with other FPGA models.

Table 2: Comparison of area and speed for different FPGA device model

DEVICE MODEL	Flip-Flops	Slices	LUTs	RAMs	IOs	bonded IOBs	Minimum period & Clock period (ns)	Maximum output required time after clock (ns)	Total REAL time to XST completion (Secs)	Total CPU time to XST completion (Secs)	Total memory usage (KB)
SPARTAN 3 XC3S400-4-PQ208	16	9 out of 3584	18 out of 7168	8	46	42 out of 141	**2.788** (1.987 logic, 0.801 route) (71.3% logic, 28.7% route)	7.165	21.00	20.73	269776
SPARTAN 6 XC6SLX150-3-FGG676	16	8 out of 184304	18 out of 92152	8	46	42 out of 396	**1.835** (1.256 logic, 0.579 route) (68.5% logic, 31.5% route)	3.819	4.00	4.23	203604
VIRTEX 4 XC4VLX200-11-FF1513	16	9 out of 89088	18 out of 178176	8	46	42 out of 960	**1.790** (1.477 logic, 0.313 route) (82.5% logic, 17.5% route)	4.221	22.00	21.84	291984
VIRTEX 5 XC5VLX110T-1-FF1136	16	8 out of 69120	18 out of 69120	8	46	42 out of 640	**1.708** (1.372 logic, 0.336 route) (80.3% logic, 19.7% route)	3.259	22.00	22.36	327952
VIRTEX 6 XC6VLX75T-1-FF784	16	8 out of 93120	18 out of 46560	8	46	42 out of 360	**1.675** (1.276 logic, 0.399 route) (76.2% logic, 23.8% route)	0.777	6.00	5.49	219348

The statistics of the FPGA resource utilized by the background subtraction module versus the pixel width of the video frame is given in Figure 8. The experimental results show that the resources utilized by object detection module are directly proportional to the pixel width and are closely linearly increased with respect to the pixels width of the frame.

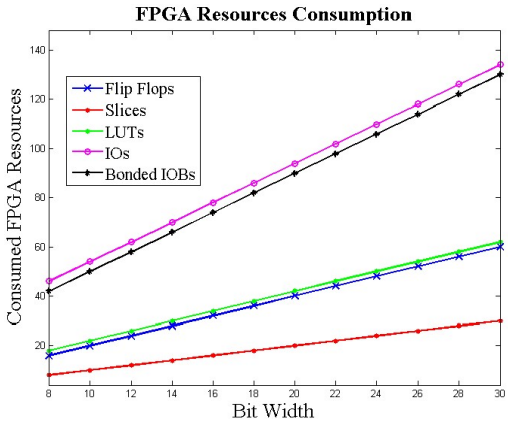

Figure 8: FPGA resources utilized by background subtraction module with respect to pixel width of the frames

When the pixel width is 8-bit in the frame, the proposed unit uses 16 Flip Flops, 8 Slices, 46 IO pads, 42 bonded IO Blocks and 18 LUTs in the hardware implementation. If 16-bit is used to represent the pixel width of frames, the proposed

surveillance unit utilizes 32 Flip Flops, 16 Slices, 78 IO pads, 74 bonded IO Blocks and 34 LUTs which shows that the resources utilized by the module are increased linearly when the pixel width of the frame is increased by the factor of 2.

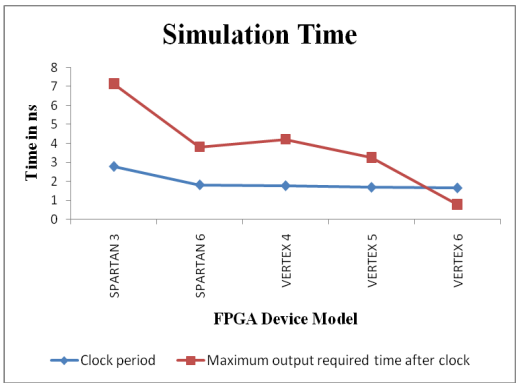

Figure 9: Comparison of simulation time for the proposed model for different FPGA devices

A comparison of the performance of the system implemented in FPGA with a more standard system shows that high speed of reconfigurable SoC can be developed for the proposed design unit as in Figure 9.

The power analysis of the whole design is carried out by the Xilinx Power Analyzer. The total power dissipated from the design unit is 1043.81mW at a temperature of 50ºC with a core voltage of 1.0 v in Virtex 5 FPGA. The maximum frequency, total power dissipation and the ambient temperature obtained from the simulation results of the FPGA implementation are presented in Table 3. The results generated for four different FPGA architectures, a Spartan 3, a Virtex 4, a Spartan 6 and a Virtex 6 are also given in Table 3 in order to show that the power consumed is different for each FGPA hardware implementation and accurate hardware design can be developed for the proposed algorithm. The Total power dissipation in Vertex 4 FPGA implementation of the proposed algorithm is reduced by 23.44% in Vertex 5 FPGA implementation as shown in Table 2. It shows that low power hardware architecture can be developed for the proposed algorithm using Vertex 5 FPGA.

Table 3: Comparison of power dissipation and maximum frequency with respect to FPGA device

Device Model	Core Voltage (V)	Process Technology	Maximum frequency (MHz)	Total power dissipation (mW)	Ambient Temperature (ºC)
SPARTAN 3	1.2	90 nm	358.680	59.84	25
SPARTAN 6	1.2	45 nm	545.004	112.79	25
VIRTEX 4	1.2	90 nm	558.659	1363.41	50
VIRTEX 5	1.0	65 nm	585.480	1043.81	50
VIRTEX 6	1.0	40 nm	597.015	1179.06	50

5. CONCLUSION AND FUTURE WORK

In this work, a reconfigurable SoC architecture for wireless video sensor network node capable of extracting a moving object in video surveillance system using background subtraction algorithm is developed. The reconfigurable SoC architecture is tested in XUPV5-LX110T Virtex-5 FPGA kit. The RS232 cable is used to interface the kit with the PC. The test results are compared with the test results obtained from a high performance device model of FPGA and are found to be satisfactory and the Total power dissipation is reduced by 23.44%. The proposed architecture is able to perform real-time moving object detection on high resolution video sequences with frame size 4608 × 3456. The proposed System processes 10fps and therefore the processing capability is 159.3 Mps. The area taken and the speed of the algorithm are also evaluated. The proposed hardware implementation is compared with the software implementation running on a 2.10 GHz Intel i3-2310M processor. The hardware implementation is found to be 26.36% faster than software implementation for complex video inputs.

In comparison with other devices, the proposed system presents a low-cost architecture. This fact shows that the implementation of this system on low-cost FPGAs is possible, and it presents good results. As future works, an Internet of Things (IoT) based reconfigurable System on Chip (SoC) that is fast and efficient for computer vision application is proposed.

REFERENCES:

[1] R. Cucchiara, G. Gualdi, "Mobile Video Surveillance Systems: An Architectural Overview," Proceedings of the first Springer international Workshop on Mobile Multimedia Processing (WMMP 2008), pp. 89-109, 2008.

[2] P. Kulkarni, D. Ganesan, P. Shenoy, Q. Lu, "SensEye: a multi-tier camera sensor network, " Proceedings of the ACM International Conference on Multimedia, pp. 229–238, 2005

[3] K. Appiah, A. Hunter, "A single-chip FPGA implementation of real-time adaptive background model," Proceedings of the IEEE International Conference on Field-Programmable Technology, pp. 95 –102, 2005.

[4] Hongtu Jiang, H. Ardo, V. Owall, "Hardware accelerator design for video segmentation with multi-modal background modeling, " Proceedings of the IEEE International Symposium on Circuits and Systems, (ISCAS'05), pp. 1142 – 1145, 2005.

[5] Jozias Oliveira, Andr´e Printes, R. C. S. Freire, Elmar Melcher, Ivan S. S. Silva, "FPGA architecture for static background subtraction in real time, " Proceedings of the 19th annual symposium on Integrated circuits and systems design. (SBCCI '06), pp. 26–31, 2006.

[6] M. Gorgon, P. Pawlik, M. Jablonski, J. Przybylo,"Pixelstreams-based implementation of video detector," Proceedings of 15th Annual IEEE Symposium on Field-Programmable Custom Computing Machines. (FCCM'07), pp. 321 –322, 2007.

[7] N. T. Vu Dang, L. N. T. My, T. N. Quoc and L. T. Hoang, "Hardware Architecture Design For Vehicle Detection," 2019 International Symposium on Electrical and Electronics Engineering (ISEE), pp. 33-36, 2019.

[8] P. G. Bhat, B. N. Subudhi, T. Veerakumar and S. Esakkirajan, "A Fully Automatic Feature-Based Real-Time Traffic Surveillance System Using Data Association in the Probabilistic Framework," in IEEE Transactions on Intelligent Transportation Systems, pp. 1–10, 2021.

[9] M.M. Abutaleb, A. Hamdy, M.E. Abuelwafa, E.M. Saad, "FPGA-based object-extraction based on multimodal sigma - delta background estimation," Proceedings of 2nd International Conference on Computer, Control and Communication. (IC4'09), pp. 1–7, 2009.

[10] Y. Tian, L. Cao, Z. Liu and Z. Zhang, "Hierarchical Filtered Motion for Action Recognition in Crowded Videos," *in* IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 3, pp. 313-323, May 2012.

[11] S. Li, J. Wu, C. Long and Y. -B. Lin, "A Full-Process Optimization-Based Background Subtraction for Moving Object Detection on General-Purpose Embedded Devices," in IEEE Transactions on Consumer Electronics, vol. 67, no. 2, pp. 129-140, May 2021.

[12] M. Benetti, M. Gottardi, T. Mayr and R. Passerone, "A Low-Power Vision System With Adaptive Background Subtraction and Image Segmentation for Unusual Event Detection," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 65, no. 11, pp. 3842-3853, Nov. 2018.

[13] L. Li, Z. Wang, Q. Hu and Y. Dong, "Adaptive Nonconvex Sparsity Based Background Subtraction for Intelligent Video Surveillance," in IEEE Transactions on Industrial Informatics, vol. 17, no. 6, pp. 4168-4178, June 2021.

[14] T. Nivetha, B. S. C. Kapali and S. P. Shally, "Video-Object Detection using Background Subtraction in Spartan 3 FPGA Kit," 2020 7th International Conference on Smart Structures and Systems (ICSSS), pp. 1-4, 2020.

[15] N. S. Sakpal and M. Sabnis, "Adaptive Background Subtraction in Images," 2018 International Conference On Advances in Communication and Computing Technology (ICACCT), pp. 439-444, 2018.

[16] P. Latha, M. A. Bhagyaveni, "Reconfigurable FPGA Based Architecture for Surveillance Systems in WSN," Proceedings of the IEEE International Conference on Wireless Communication and Sensor Computing (ICWCSC'10), pp. 1–6, 2010.

[17] P. Latha, M. A. Bhagyaveni, J. Ancilin, "VLSI Implementation of Reconfigurable WSN Node for Surveillance," European Journal of Scientific Research, vol. 92, no. 2, pp. 238-256, 2012.

[18] P. Latha, M. A. Bhagyaveni, Steffi Lionel, "Reconfigurable SoC Architecture for Ship Intrusion Detection," Journal of Theoretical

and Applied Information Technology, vol.60, no.1, pp. 95-105, 2014.

[19] A. Lopez-Bravo, J. Diaz-Carmona, A. Ramirez-Agundis, A. Padilla-Medina, J. Prado-Olivarez, "FPGA-based video system for real time moving object detection," Proceedings of the IEEE International Conference on Electronics, Communications and Computing (CONIELECOMP'13), pp. 92-97, 2013.

[20] C. S´anchez-Ferreira, J. Y. Mori, C. H. Llanos, "Background Subtraction Algorithm for Moving Object Detection in FPGA," Proceedings of the IEEE VIII Southern Conference on Programmable Logic (SPL'12), pp. 1–6, 2012.

[21] S. H. Ahmed, S. Y. Elhabian, El-Sayed K. M, "Moving Object Detection in Spatial Domain using Background Removal Techniques - State-of-Art," Recent Patents on Computer Science, vol.1, pp. 32–34, 2008.

[22] Shahriar Negahdaripour, "Revised Definition of Optical Flow: Integration of Radiometric and Geometric Cues for Dynamic Scene Analysis," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.20, no.9, pp. 961-979, 1998.

[23] Simon Baker, Daniel Scharstein, J.P. Lewis, Stefan Roth, Michael J. Black, Richard Szeliski, "A Database and Evaluation Methodology for Optical Flow," International Journal of Computer Vision. Vol.92, no.1, pp. 1–31, 2011.

[24] Z. Chaohui, D. Xiaohui, X. Shuoyu, S. Zheng, L. Min, "An improved moving object detection algorithm based on frame difference and edge detection," Proceedings of the 4th Int. Conf. Image Graph, pp. 519–523, 2007.

[25] R. Mingwu, S. Han, "A practical method for moving target detection under complex background," Comput. Eng, vol. 31, no. 20, pp.33–34, 2005.

[26] X. Benxian, L. Cheng, C. Hao, Y. Yanfeng, C. Rongba, "Moving object detection and recognition based on the frame difference algorithm and moment invariant features," Proceedings of the 27th Control Conf. pp. 578–581, 2008.