



DESIGN AND IMPLEMENTATION OF FPGA ACCELERATOR FOR VISION BASED FIRE DETECTION

A. Hassan*, C. U. Ngene and P. Y. Dibal

Department of Computer Engineering, University of Maiduguri, Borno State, Nigeria.

*Corresponding author's e-mail address: abubakarhassan@unimaid.edu.ng

ARTICLE INFORMATION

Submitted: 15th August 2024
Revised: 27th August 2024
Accepted: 28th November 2024

Keywords:

FPGA
Hardware accelerator
Simulation
Image processing
Fire detection

ABSTRACT

Fire outbreaks are great threat to human beings, economic infrastructure, and the environment. This led to the need for timely and accurate detection of fire incidents to minimize their devastating impacts. But conventional fire detection systems are electronic sensor based which as result they suffer from the problems of transport delay, conduction delay, limited detection range, high false alarm, and inappropriate for outdoor applications. To address the shortcomings of such sensor-based methods, several image processing and computer vision approaches to fire detection have been proposed. However, these image processing and computer vision-based solutions are implemented in software platforms which have disadvantages of inefficiency, high hardware requirements and high cost. In this research work, an embedded hardware accelerator for vision-based fire detection was designed and implemented in Kintex-7 series Field Programmable Gate Array (FPGA). MATLAB R2021a software was used for decoding the image dataset into pixel stream data. The design was captured in very high-speed integrated circuit HDL (VHDL). The design was synthesized with Xilinx Vivado 2021 design suite and simulated with Xilinx ISIM. Evaluation results showed that the accelerator could achieve good detection of fire features and were consistent with the results from MATLAB code running on personal computer. Compared to software implementation, the latency, resource utilization and power consumption was greatly reduced. It was also found that the hardware accelerator which was developed as an Intellectual Property (IP) core can also be employed to speed up grayscale conversion, edge detection and thresholding algorithms in embedded vision, smart camera and video analytics applications.

© 2025 Faculty of Engineering, University of Maiduguri, Nigeria. All rights reserved.

1.0 Introduction

Fire emergencies are leading to huge human loss, massive infrastructure damage and serious environmental degradation around the world. For instance, the Kano State Fire Service records that 134 human lives and 635-million-naira assets were lost due to fire outbreaks in the state between January 2021 to December 2021. Another fire event was also reported at Sabon Gari Market in Kano City in which properties and shops worth millions of Naira were lost and took more than 10 hours to be put out completely by the Kano State Fire Service team. Report from News Agency of Nigeria (NAN) on 23 May 2021, showed that not less than 64 people, including eight fire fighters, got different degrees of injuries from a petroleum tanker explosion at a filling station in Sharada area of Kano State (Hassan & Audu, 2022; Rabi, 2022). This necessitates the need for timely and accurate fire detection systems to minimize their catastrophic impact of fire disasters. But majority of existing fire detection systems are electronic sensor based and have problems of transport delay, conduction delay, limited detection range, high false alarm, inappropriate for outdoor applications and lack of capability to give rich additional information on location, growth rate, size of the fire incidents. (Hassan & Audu, 2022; Rabi, 2022; Solórzano et al., 2022; Wang et al., 2020). Several image processing and computer vision-based methods have been proposed to solve the problems of such sensor-based methods. However, majority of these image processing and computer vision-based approaches were implemented in software

which as a result have disadvantages of high hardware requirement, high cost and inefficiency (Abid et al., 2021; Choudhuri & Roy, 2016; Liu et al., 2020).

The above problems can be effectively using hardware acceleration technology platforms such as FPGA. FPGA has inherent advantages of latency and higher energy consumption for streaming computing tasks. Furthermore, FPGA has high flexibility which can change hardware algorithms and chip functions to facilitate algorithm and function verification (Hassan et al., 2024).

This paper aims to propose FPGA hardware accelerator for vision-based fire detection applications implemented in VIVADO and MATLAB environments. The hardware accelerator consists of an image decoding module, grayscale conversion module, edge detection module and fire pixel classification module. The remaining part of this paper is organized as follows. Section 2 presents review of related works methodology. Section 3 presents the methodology. Results and discussion are provided in section 4. Finally, a conclusion and future direction is presented in section 4.

Several image processing and computer vision approaches to fire detection have been proposed. Authors in (Pritam & Dewan, 2017) proposed fire detection system based on LUV color space and hybrid transforms. Video frames captured are read and are segmented. All frames are analyzed to confirm the presence of fire. Fire is detected by the presence of fire pixels in the image. Fire pixels are recognized using the features of LUV color space and hybrid transforms. Edge detection algorithm is employed to extract the edges of the image when flame color detects existence of fire. Image segmentation methods are employed to extract the fire region of interest from the background. An algorithm for smoke detection based on simplified descriptors of video information is proposed by (Monte et al., 2017). Firstly, the descriptors of the scene on every frame are obtained. The inference processes of smoke presence are input by the descriptors and background estimation, which is updated with every frame. A group of classifiers perform the smoke inference process. The fusion outcomes of these classifiers provide a reliable detection. Simple Boolean logic combines the process outcomes, with minimum and maximum threshold values. The proposed approach is simple and efficient for deployment on computationally low embedded applications for real-time surveillance.

Researchers in Vijayalakshmi & Muruganand, (2018) proposed an effective smoke detection approach in video images for early fire emergency systems. The proposed algorithm uses fuzzy c-means clustering and background subtraction to segment smoke region of interest from moving regions. Result obtained show that the algorithm outperforms other algorithms and increase the accuracy smoke detection. However, the algorithm is not tested on large variety of dataset and the performance still can be improved because it is only based on motion segmentation. Zhang et al. (2017) developed a method for smoke detection in videos. The proposed algorithm employs two key features of smoke to detect it. Visual background extractor (ViBE) is employed to segment moving objects in static scene. The atmospheric model is used to detect semi-transparent feature of smoke. The similarity between extracted regions and original background regions in the same position is computed using normalized cross correlation and the value of the original background is chosen from the frame that is nearest to the current frame. The blocks with high similarity are identified as smoke parts. Experiment results demonstrate that the algorithm can recognize fire and integrated into surveillance system. Authors in Huang et al., (2017) proposed an algorithm for video-based smoke detection in low illumination indoor environment using support vector machine. The smoke region is extracted and then the area diffusion and main direction of movement of smoke region are selected as the dynamic features and texture is extracted as the static feature to recognize fire smoke in low-light indoor scene. Support vector machine is employed to design the classifier to realize smoke detection.

An effective method for real-time fire detection based on video processing is proposed by (Khan et al., 2019). The proposed method employs flame color and spatial-temporal features to detect fire regions. Firstly, fire color pixels are segmented using a set of improved rules based on RGB color space. The foreground is segmented based on change in blue component trained using neural network. Lastly, the region of interest is extracted using growing area analysis. Authors in Yang et al., (2018) proposed technique for video-based

smoke detection by combining Gaussian Mixture Model (GMM) and HSV color model with the deep convolutional neural network. First, each frame is divided into 10-by-10 sub-images by the method. Secondly, Gaussian Mixture Model is employed to extract the motion region and later the sub-images block which contains the moving region is segmented. Then, the motion blocks segmented by the GMM is used to extract target smoke blocks successively using HSV color feature analysis. Zeng et al. (2018) proposed an effective method for smoke detection based on automatic vision system. The method employs an adaptive background subtraction technique to segment moving regions. Non-smoke regions are extracted out by employing color feature and fuzzification. Regions that satisfy both the color analysis and fuzzification criteria are classified as candidate smoke regions. In (Kaabi et al., 2018), a novel approach for video-based smoke detection based on machine learning approach to overcome forest wildfires was proposed. Firstly, the video is divided into frames for training with deep belief network. In addition, Gaussian Mixture Model is employed to segment candidate regions from the target smoke video to be trained by using Deep Belief Network. Logistic regression is used to classify on the output data. Researchers in (Xie et al., 2020), proposed a method that employs the motion-flicker based dynamic features and deep static features for video-based fire detection. First, the dynamic features are segmented by, analyzing the differences in motion and flicker features between fire and non-fire objects in video scenes. Second, an adaptive lightweight convolutional neural network is proposed to segment the deep static features of fire. Zhang et al., (2021) developed an effective asymmetric encoder-decoder U-shape architecture based on Squeeze Net, Attention U-Net and Squeeze Net performs most of the time as an extractor and a discriminator of forest fire. The model uses attention mechanism to extract useful features and suppress non-related features by embedding Attention Gate (AG) units in the skip connection of U-shape structure.

A robust algorithm based on information on color and shape was proposed by (Rabiu, 2022). The algorithm starts with loading video image database developed to detect presence or absence of fire in video frames. Background subtraction is used by this method to compare the current frame with the reference frame. When the outcome of the background subtraction is less than the set point, the frame is discarded, and the next frame is checked. But if the difference is equal or greater than the set point, the frame is then subjected to color and shape test. This is achieved by employing combined RGB color space model and shape feature. The proposed algorithm was extremely robust in detecting fire compared to those algorithms based on color or motion features.

2. Materials and Methods

The architecture of the FPGA hardware accelerator is presented in this section. It mainly consists of image decoding module, grayscale image conversion module, Sobel edge detection and fire pixel classification module. Modular and hierarchical approach was adopted for this design and the model was captured using VHDL. VHDL was used in view of its strength in paper documentation. The block diagram of the hardware accelerator is depicted in Figure 1.

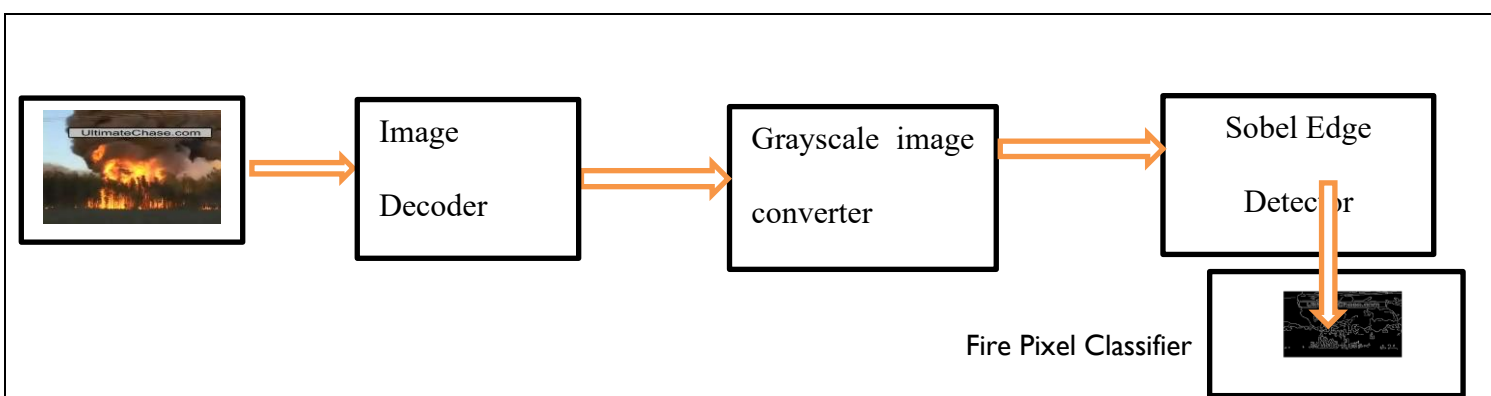


Figure 1: Block Diagram of the Hardware Accelerator.

2.1 Image Decoding

Video dataset used in this work was obtained from VisiFire database which is a commonly used public video dataset for fire and smoke detection (Hassan et al., 2024). Edge pixel features are used in classifying objects into fire or non-fire classes, because edges preserve structural information of objects.

The FPGA accelerator read video frames from Block RAM and convert the images into streams of pixel data for pixel streaming computation after obtaining average RGB pixel values in MATLAB. The pseudocode for converting video data in frames is shown as follows.

Pseudocode for Video Frames conversion.

- 1: READ video data stream;
- 2: COMPUTE number of frames in the video stream;
- 3: FOR i = 1:2: n
 - CONVERT video stream into frames
 - SAVE frames
- 4: END.

Then we computed the respective average RGB values to obtain representative video frames for pixel stream data generation. The Pseudocode for image decoding into pixel streams is depicted as follows.

Pseudocode for Image Decoding.

- 1: READ video frames:
- 2: COMPUTE number of frames;
- 3: FOR k = 1: numFrames
 - DECODE frames into pixel stream
 - SAVE pixel stream
- 4: END.

The algorithmic state machine (ASM) chart for the image decoding into pixel streams is illustrated in Figure2. It describes how the image decoding module reads and convert video frames from memory to generate the pixel streams.

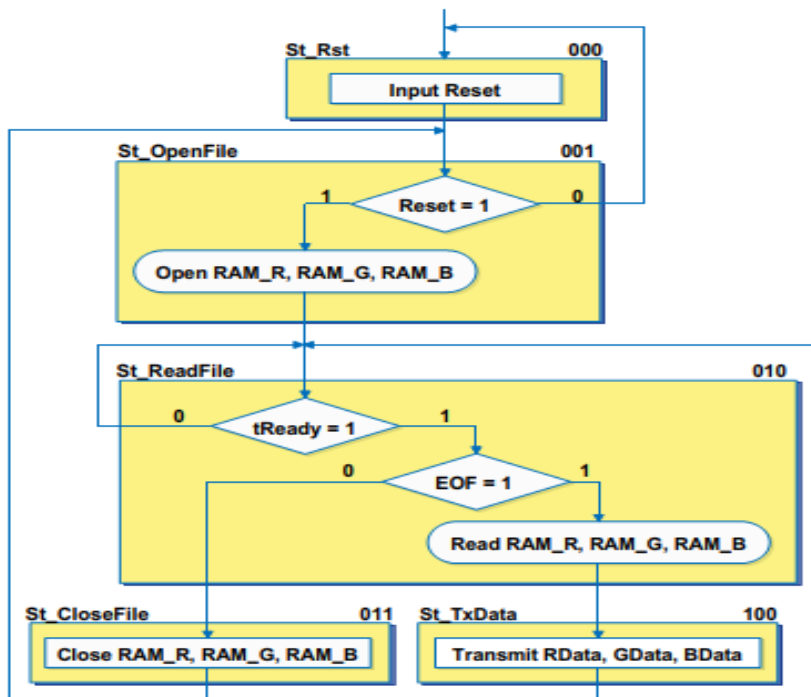


Figure 2: ASM chart for Image Pixel Generation.

2.2 Gray Scale Image Conversion

The RGB image pixel data was converted to grey scale image for easy computation which is expressed as given by in equation (1).

$$Y = 0.299R + 0.587G + 0.111B \quad (1)$$

For the hardware implementation, we avoided the floating-point multiplications and efficiently computed the grayscale value of the pixel as given by in equation (2).

$$Y = 0.299R + 0.587G + 0.114B \approx \frac{306 \times R + 601 \times G + 117 \times B}{1024} \quad (2)$$

Figure 3.3 illustrates the architecture of the RGB to Gray scale image converter.

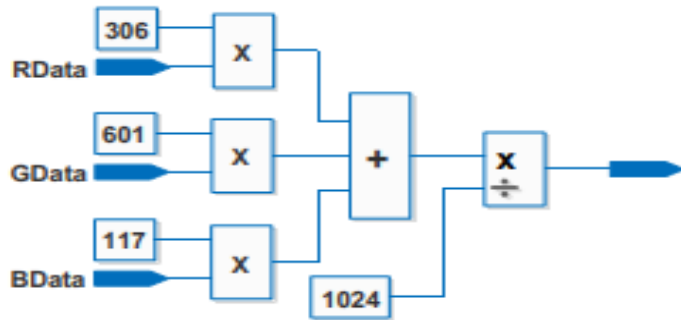


Figure 3. Architecture of the RGB to Grayscale Image Converter

2.3 Sobel Edge Detection

The Sobel edge detection module is responsible for extracting and classifying object edges. It also writes the result the detected edges array into the fire pixel classification module. Sobel edge detection is a classical algorithm in the field of image and video processing for the extraction of object edges. Edge detection using Sobel operators works on the premise of computing an estimate of the first derivative of an image to extract edge information. By computing the x and y direction derivatives of a specific pixel against a neighbourhood of surrounding pixels, it is possible to extract the boundary between two distinct elements in an image. The operator consists of a pair of 3 by 3 convolution kernels as shown in equation (3). one kernel is simply the other rotated by 90 degrees.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * A \quad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -2 \end{bmatrix} * A \quad (3)$$

Where A is the input image, G_x, G_y are images where each point represents the approximate derivatives along the x and y axes.

We assume that the original image is given by in equation (3.4).

$$\begin{bmatrix} f(x-1, y-1) & f(x, y-1) & f(x+1, y-1) \\ f(x-1, y) & f(x, y) & f(x+1, y) \\ f(x-1, y+1) & f(x, y+1) & f(x+1, y+1) \end{bmatrix} \quad (4)$$

The vertical and horizontal gradients of the Sobel operators are G_x and G_y, which denote the gray level of the edge detection image in vertical and horizontal directions respectively which are computed as given by (Zhang, 2018) in equation (5) and (6)

$$G_x = Sobel_x \times A = f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1) - f(x+1, y-1) - 2f(x-1, y) - f(x-1, y+1) \quad (5)$$

$$G_y = Sobel_y \times A = f(x-1, y+1) + 2f(x, y+1) + f(x+1, y+1) - f(x-1, y-1) - 2f(x-1, y) - f(x+1, y-1) \quad (6)$$

These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The approximate gradient value of brightness can be obtained by element-wise summation of the resulting images given by in equation (7).

$$|G| = \sqrt{[G_x^2 + G_y^2]} \quad (7)$$

Typically, an approximate magnitude is computed using equation (3.8) .

$$|G| = |G_x| + |G_y| \quad (8)$$

2.4 Fire Pixel Classification

In this research work, we employed thresholding-based technique to detect fire edge pixels which are expressed as given by in equation (9)

$$g(x, y) = \begin{cases} \text{Fire,} & \text{If } f(x, y) \geq TH \\ \text{Not Fire,} & \text{Otherwise} \end{cases} \quad (9)$$

Where $f(x, y)$ is the input image and TH is the threshold value.

The threshold value was determined using the Otsu’s method which is used to perform automatic image thresholding in computer vision and image processing. The pseudocode for the Otsu method is given as follows.

Pseudocode for Otsu Method

- 1: COMPUTE the normalized histogram of the input image.
- 2: COMPUTE the cumulative mean.
- 3: COMPUTE the global intensity mean.
- 4: COMPUTE the between-class variance.
- 5: OBTAIN the optimum threshold.
- 6: SEGMENT the image using the threshold.

3. Results and Discussion

The results of the actual design of the FPGA hardware accelerator are first presented by examining the synthesis and the simulation results.

3.1 Synthesis

The hardware was synthesized with Xilinx Vivado 2021.1 design suite and simulated using Xilinx ISIM. Test benches were used to verify the functionality of the design, which was implemented in Kintex-7 series FPGA by using Xilinx configuration tools. The device utilization summary for the targeted Kintex-7 is shown in Figure 4.

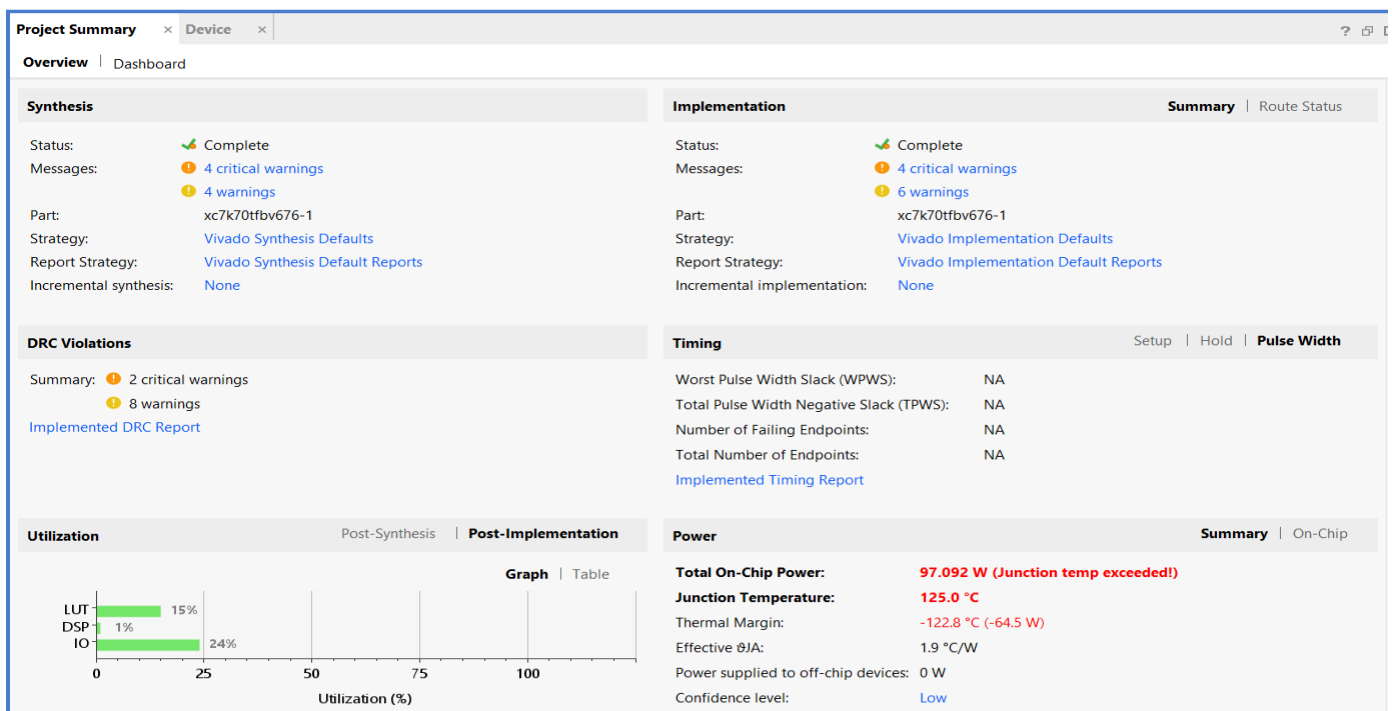


Figure 4: Device Utilization summary

It is evidently clear from the summary that some ports of the devices were not efficiently utilized, especially the number of DSPs. The motivation behind the use of Kintex-7 was because it is robust, highly reconfigurable and have lower power consumption. The input/output lines were large due to the high level of parallelism of the design, which generally enhanced the speed of operation of the device at the expense of area or huge number of primitives employed.

3.2 Verification of Results

The test benches used to verify the functionality of the design is shown in the Appendix. The FIRESourceSignal entity was instantiated to provide RGB image data. The test stimuli were driven to the FIRESourceSignal inputs. The simulation in Figure 5 shows how the image decoding module read that the RGB image frames simultaneously from memory and decoded them into pixel stream data array at the rising edge of the clock. Then the grayscale conversion module read the RGB pixel stream data array and converted into grayscale image concurrently. The result of the grayscale conversion is depicted in Figure 7. The result of Sobel Magnitude Computation is depicted in Figure 6 and that of Sobel Square Root Computation is shown in Figure 7.

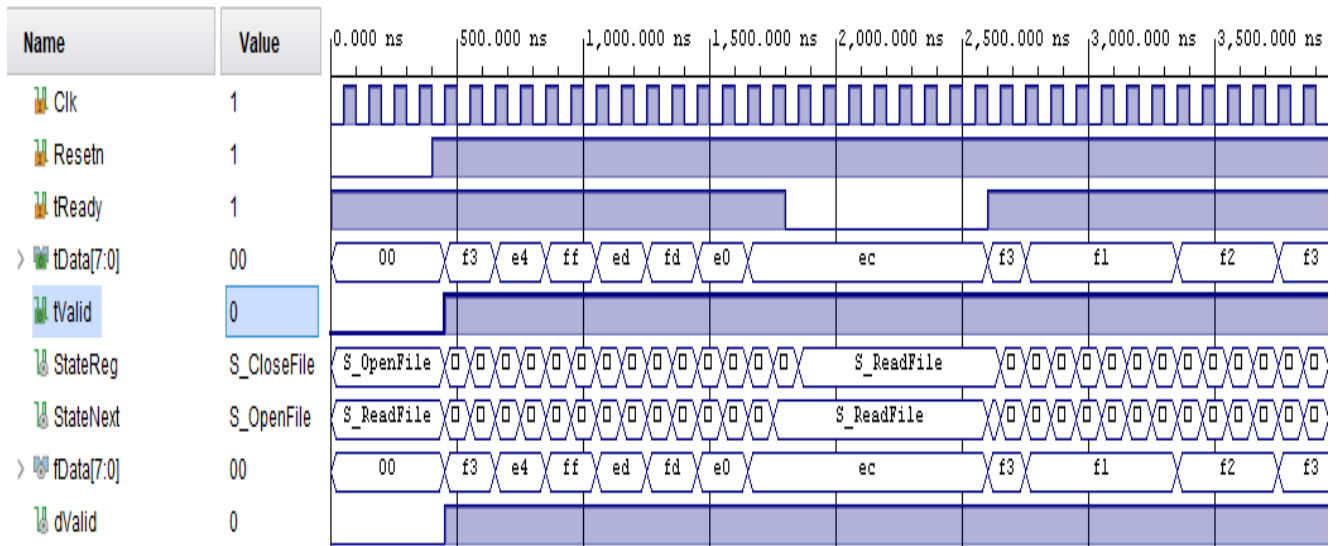


Figure 5: Image Pixel Generation Verification Waveforms

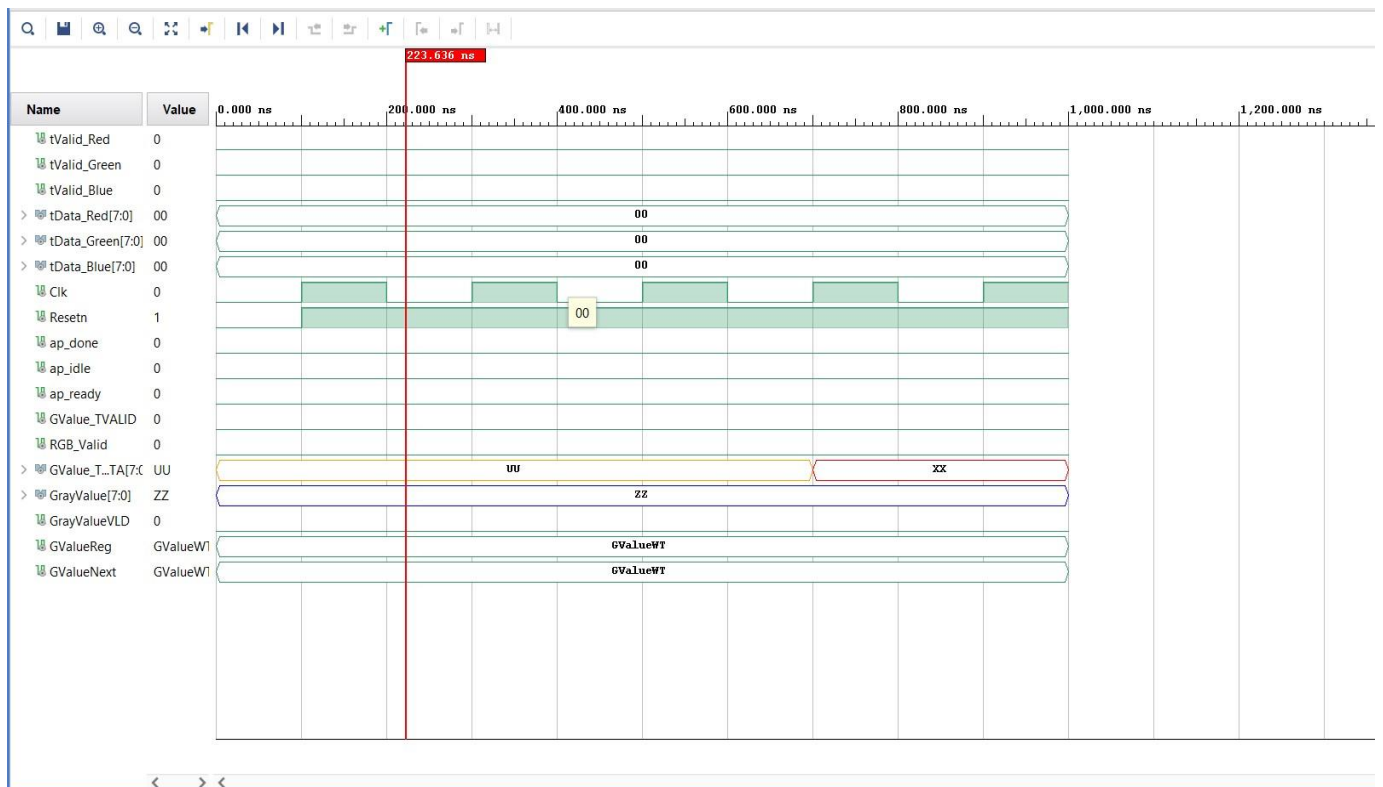


Figure 6: Grayscale Image Conversion Verification Waveforms

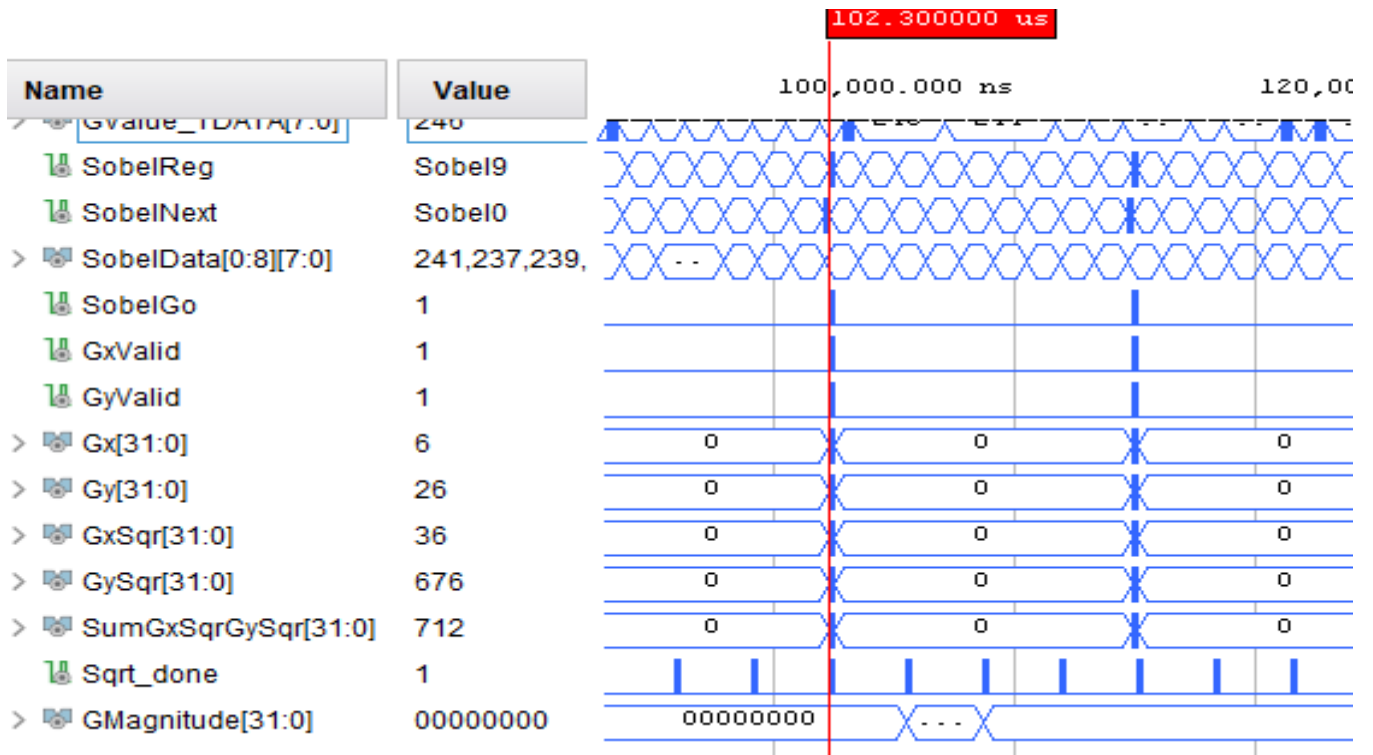


Figure 7: Sobel Magnitude Computation Verification Waveforms

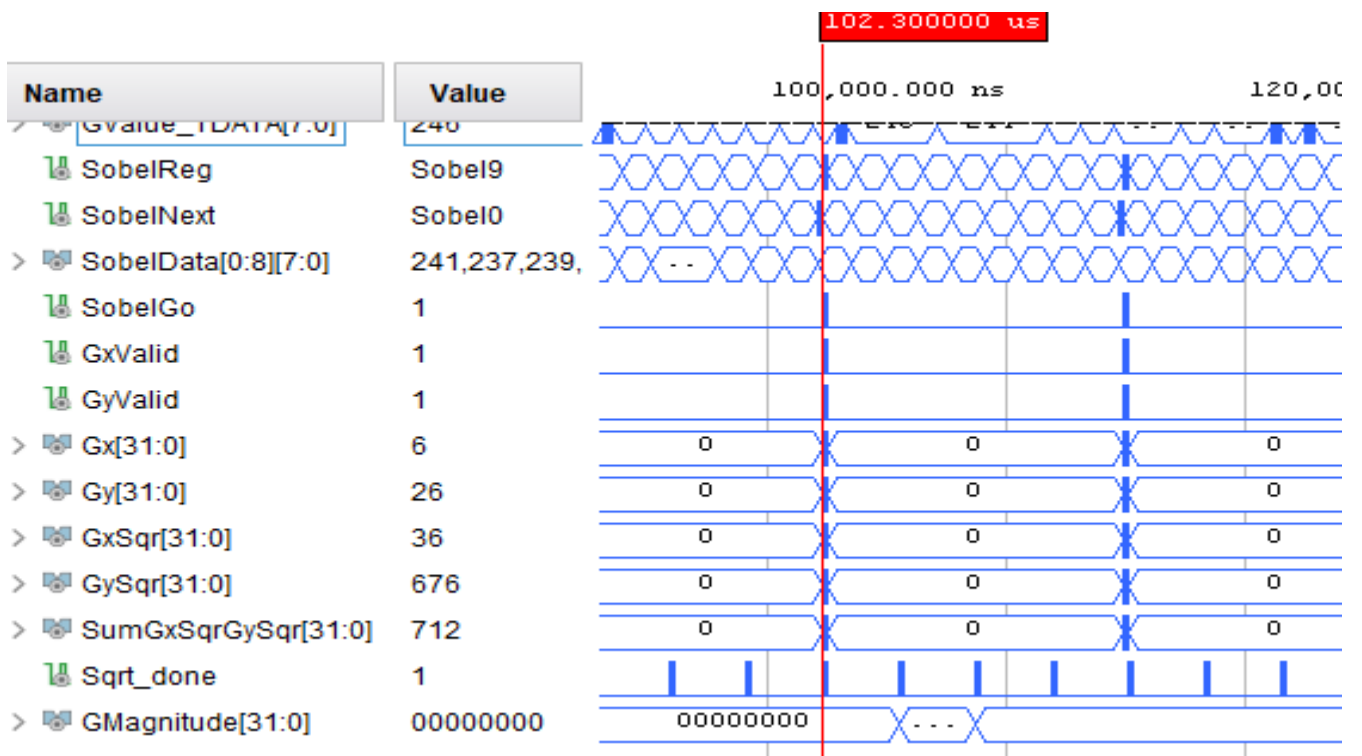


Figure 8: Sobel Square Root Computation Verification Waveforms

3.3 FPGA Resource Utilization

The Hardware accelerator consumes 15% of available LUTs (look up table), 24% of IO (Input-Output) and 1% DSPs. The specific hardware utilization in comparison with a similar work by authors in (Choudhuri & Roy, 2016) is presented in Table I. The accelerator has a shorter processing time and operate up to more than 200 MHz.

Table I. Resource Utilization and Performance Comparison

Author and Date	LUT (%)	IO (%)	DSP (%)	Latency(ms)
Choudhuri & Roy, 2016.	1	7	1	3.2
Our work	15	24	1	2.97

It is evident that the FPGA accelerator has efficient resource utilization, and the input/output lines were large due to the high level of parallelism of the design, which generally enhanced the speed of operation of the device at the expense of area or huge number of primitives employed. The power consumption of the hardware accelerator is illustrated in Figure 9. The dynamic power consumption is due to the switching between digital states and the static power consumption is because of leakages in the circuit.

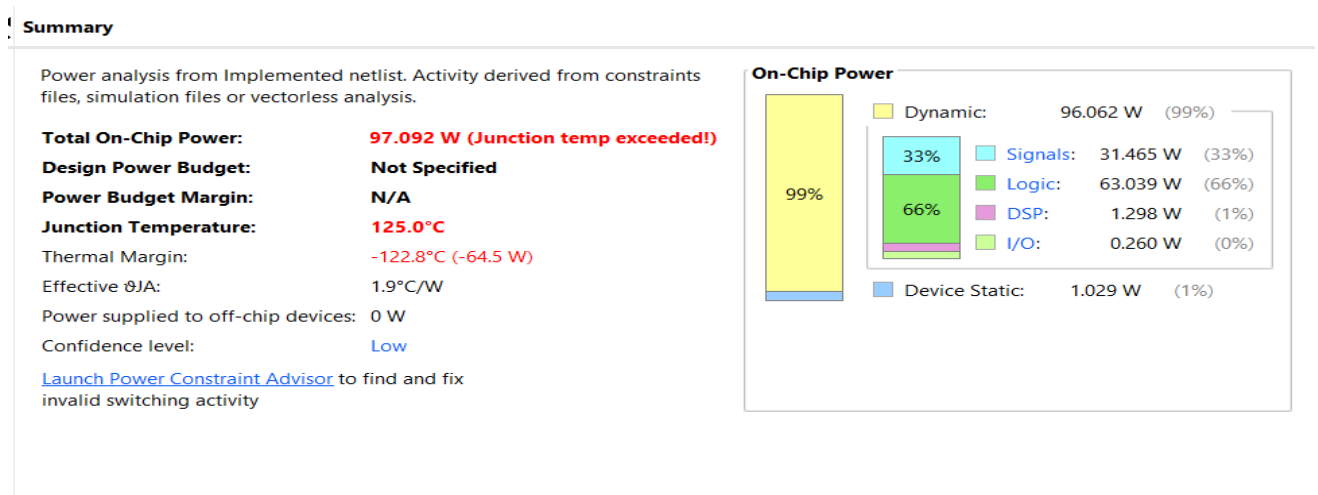


Figure 9. Power Analysis of the Design

4. Conclusion

This research work was embarked upon to develop FPGA hardware accelerator for vision-based fire detection which could realize the acceleration of image decoding, grayscale conversion and edge detection. The design was verified and implemented using simulation, synthesis and configuration tools of Xilinx Inc. USA and MATLAB. The accelerator performs image decoding, grayscale conversion, Sobel edge detection and pixel thresholding.

The FPGA accelerator was implemented Xilinx kintex-7 FPGA. The evaluation results based on the test benchmark dataset show that the accelerator could achieve good detection of fire edge features and were consistent with the processing results of the MATLAB code on personal computer. It consumes 15% Look-up Table (LUT), 1% Digital Signal Processor (DSP), 24% Input-Output (IO), 92.062W Dynamic and 1.029W Device Static power. The hardware accelerator which was developed as an Intellectual Property (IP) core can also be employed to perform image decoding, grayscale conversion, edge detection and thresholding in embedded vision, smart camera and video analytics applications.

The following areas are recommended for future research.

- 1) Interfacing camera and display monitor to the hardware accelerator for real-time fire detection.
- 2) Employing advanced image processing and computer vision algorithms such as running average background subtraction, canny edge detection, etc.
- 3) Designing embedded machine learning accelerator for fire outbreak prediction.
- 4)

Acknowledgement

This study was supported by Petroleum Technology Development Fund (PTDF) with grant number PTDF/ED/LSS/MS/AH/0657/19.

References

- Abid, F., Izeboudjen, N., & Louiz, F. (2021). Hardware/Software Development of a Machine Learning based Forest Fires Prediction System. *Proceedings - 2021 International Conference on Information Systems and Advanced Technologies, ICISAT 2021*. <https://doi.org/10.1109/ICISAT54145.2021.9678486>
- Aqeel Khan, M. N., Tanveer, T., & Khurshid, K. (2019). Fire Detection System using Raspberry Pi. *2019 International Conference on Information Science and Communication Technology (ICISCT)*.
- Choudhuri, A., & Roy, P. (2016). VLSI architecture for real time fire detection. *2015 IEEE International Conference on Computer Graphics, Vision and Information Security, CGVIS 2015*, 27–32. <https://doi.org/10.1109/CGVIS.2015.7449887>
- Hassan, A., & Audu, A. I. (2022). Traditional Sensor-based and Computer Vision-based Fire Detection Systems: A Review. *Arid Zone Journal of Engineering, Technology and Environment (AZOJETE)*, 18(3), 469–492. www.azojete.com.ng
- Hassan, A., Umerah Ngene, C., & Yusuf Dibal, P. (2024). Performance Evaluation of FPGA Accelerator for Vision based Fire Detection System. *Bayero Journal of Engineering and Technology (BJET)*, 19(2).
- Huang, M., Wang, Y., & Hu, Y. (2017). Smoke Identification of Low-Light Indoor Video Based on Support Vector Machine. *IEEE*.
- Kaabi, R., Sayadi, M., Bouchouicha, M., Fnaiech, F., Moreau, E., & Ginoux, J. M. (2018). Early smoke detection of forest wildfire video using deep belief network. *2018 4th International Conference on Advanced Technologies for Signal and Image Processing, ATSIP 2018*, 1–6. <https://doi.org/10.1109/ATSIP.2018.8364446>
- Liu, Y., Sun, R., Zhang, T., Zhang, X., Li, L., & Shi, G. (2020). Fast Fire Identification Soft-Core Package Design Based on FPGA. *2020 International Conferences on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics)*, 642–647.
- Monte, G., Pastore, J. I., Bailarin, V., Marasco, D., & Liscovsky, P. (2017). Smoke detection using simplified descriptors of video information. *Proceedings of the IEEE International Conference on Industrial Technology*, 1070–1075. <https://doi.org/10.1109/ICIT.2017.7915510>
- Pritam, D., & Dewan, Jaya. H. (2017). Detection of fire using image processing techniques with LUV color space. *2017 2nd International Conference for Convergence in Technology (I2CT)*.
- Rabiu, H. (2022). Robust Fire Detection Algorithm using Color and Shape Information. *Bayero Journal of Engineering and Technology (BJET)*, 17(1), 2022–2044.
- Solórzano, A., Eichmann, J., Fernández, L., Ziems, B., Manuel Jiménez-Soto, J., Marco, S., & Fonollosa, J. (2022). Early fire detection based on gas sensor arrays: Multivariate calibration and validation. *Sensors & Actuators: B. Chemical*, 352. <https://doi.org/10.5281/zenodo.5643074>
- Vijayalakshmi, S. R., & Muruganand, S. (2018, December 28). Fire recognition Based on Sensor node and Feature of Video Smoke. *International Conference on Advanced Computation & Telecommunication: ICACAT*.
- Wang, G., Li, J., & Zheng, Y. (2020). Forest smoke detection based on deep learning and background modelling. *2020 IEEE International Conference on Power, Intelligent Computing and Systems: ICPICS 2022*.
- Xie, Y., Zhu, J., Cao, Y., Zhang, Y., Feng, D., Zhang, Y., & Chen, M. (2020). Efficient video fire detection exploiting motion-flicker-based dynamic features and deep static features. *IEEE Access*, 8, 81904–81917. <https://doi.org/10.1109/ACCESS.2020.2991338>
- Yang, Z., Shi, W., Huang, Z., & Yin, Z. (2018). Combining Gaussian Mixture Model and HSV Model with Deep Convolution Neural Network for Detecting Smoke in Videos. *18th International Conference on Communication Technology (ICCT)*.
- Zeng, J., Lin, Z., Qi, C., Zhao, X., & Wang, F. (2018). An Improved Object Detection Method based on Deep Convolution Neural Network for Smoke Detection. *2018 International Conference on Machine Learning and Cybernetics (ICMLC)*, 1.
- Zhang, J., Zhu, H., Wang, P., & Ling, X. (2021). ATT Squeeze U-Net: A Lightweight Network for Forest Fire Detection and Recognition. *IEEE Access*, 9, 10858–10870. <https://doi.org/10.1109/ACCESS.2021.3050628>

Zhang, Q., Liu, F., Li, X., & Li, B. (2017). Dissipation function and ViBe based smoke detection in video. Proceedings - 2017 2nd International Conference on Multimedia and Image Processing, ICMIP 2017, 2017-January, 325–329. <https://doi.org/10.1109/ICMIP.2017.62>

Zhang, S. (2018). Real Time Image Processing on FPGAs. University of Liverpool.