



Proposal / Application

for

Final Year Project

Computer & Information Systems Engineering Department

Software-defined Camera for Outdoor Surveillance Applications

Soniya Shafi (CS-18133)

Misha Akram (CS-18118)

Iqra Irfan (CS-18123)

Hoor Soomro (CS-18110)

NED University of Engineering & Technology

1. Project Identification

A. Reference Number (for office use only)

C	S	-	1	8		
---	---	---	---	---	--	--

B. Project Title

Software-defined Camera for Outdoor Surveillance Applications

C. Project Internal Advisor

Name	FAKHRA AFTAB
Designation	Lecturer

D. Project Internal Co-Advisor

Name	Dr. MAJIDA KAZMI
Designation	Assistant Professor

E. Project External Advisor

Name	Dr. ARSHAD AZIZ		
Designation	Professor		
Organization	NUST		
Mobile #		Email:arshad@nust.edu.pk	

F. Student Team

S. No.	Roll No.	Name	Email
1.	CS-18133	Soniya Shafi	aa9391438@gmail.com
2.	CS-18118	Misha Akram	mishaakrambaig@gmail.com
3.	CS-18123	Iqra Irfan	syedaiqrairfan950@gmail.com
4.	CS-18110	Hoor Soomro	hoorksoomro2@gmail.com

G. Sponsoring Organization

No Sponsoring Organization

H. Keywords

Edge computing, Image sensor, IP cores, High Level Synthesis (HLS)

I. Project Idea

☐ New ☒ Modification to a previous project

☐ Extension of a previous project

2. ABSTRACT

Image processing applications are widely used nowadays which need to analyze and process a large number of images and video streams in real-time. This real-time constraint can be handled if the live camera feed is preprocessed right at the edge device i.e. an FPGA. The FPGA coupled with an image sensor is used to develop a camera that is adaptive and can preprocess the incoming live camera feed using high-level synthesis. Thus, this approach proposes to define a Software-Defined Camera that utilizes the software-hardware co-design functionality of the Zynq-7000 FPGA.

3. Project Background and Literature Review

Traditional cameras are not flexible and also unaware of their environment. They cannot provide constant and standard image quality in every environment because of their fixed (non-programmable) internal architecture. Along with image quality, image pre-processing is also crucial. The images need to be pre-processed before they can be used to generate useful results. Therefore ample amount of work done only to improve image quality and making pre-processing faster.

The authors of [1] use Field Programmable Gate Array (FPGA) architecture for making pre-processing faster. As the FPGA architecture has the ability to perform parallel processing, it will shorten the processing time and the efficiency will increase.

The authors of [2] present their contribution on the noise reduction problem by proposing an intermediary step between the image sensor and the post-processing software in the image capturing process. Their solution matches the discrete samples between multiple frames and averages the pixel values. The output image maintains its structural integrity, holds better color accuracy and incurs less noise than others.

The pre-processing generally occurs at server side. Server solution is simple and effective but it is costly and time consuming. Edge computing eliminates the need to send image data to server-side for processing. Provides a way to process the images directly on the camera by taking advantage of sensors and the use of end devices to take over the load of processing [3].

In [4] the detection and recognition tasks for surveillance are executed locally by edge devices. Only when devices are not able to execute the recognition task, a recognition request is sent to the server.

The properties of FPGA (Field Programmable Gate Array) make them a desired technology to implement digital image and video processing. HLS (High Level Synthesis) Vivado is one of the most used tools which can directly transform a C description to a hardware IP, described on Verilog or VHDL. FPGA reconfigurable circuit with its HLS software tools is become a very promising technology to be widely used in many applications encompassing all aspects and requirements of embedded system [5].

In [6] authors state that, escalating system-on-chip design complexity is pushing the design community to raise the level of abstraction beyond register transfer level. Despite the unsuccessful adoptions of early generations of commercial high-level synthesis (HLS) systems, we believe that the tipping point for transitioning to HLS methodology is happening now, especially for field-programmable gate array (FPGA) designs. The latest generation of HLS tools has made significant progress in providing wide language coverage and robust compilation technology, platform-based modeling, advancement in core HLS algorithms, and a domain-specific approach.

In [7] authors survey 46 papers and 118 applications and conclude that, the QoR (Quality of Result) of RTL flow is still better than that of the state-of-the-art HLS tools. However, the average development time with HLS tools is only a third of that of the RTL flow, and a designer obtains over four times as high productivity with HLS.

In [8] authors presents four HLS design examples, including a multiplexer, counter, register block and a skin detection image processing algorithm. Each design has been implemented and tested in FPGA hardware using the Vicilogic automation and proto-typing tools developed by the authors. They state that HLS reduces the effort of RTL HDL design capture and debug. HLS also allows flexibility in the final hardware implementation in order to meet design constraints.

4. Motivation and Need

All the computer vision algorithms that perform image analysis and processing require high quality images. The conventional cameras available now a days have fixed internal architecture i.e.; they are not programmable; hence not aware of their environment and non-adaptive. These cameras can't provide a constant and standard image quality in different scenarios.

All real time applications need to process data as fast as possible; the images need to be pre-processed as well before they can be used to generate useful results according to the requirements of an application. If the environment diverges from normal conditions or noise is generated, conventional cameras fail to maintain the standard image quality. As a result, more computational intensive & time consuming algorithms will be required to generate clean and clear images before using them in computer vision applications.

This leads to the need for a software-defined camera; where imaging sensor inside camera is adaptive to respond environment. Therefore, processing the images directly on the camera, taking advantage of the sensors to become environmentally conscious. Such a camera takes over the load of preprocessing images. The strategy is to make our camera smart enough so that it can sense its surroundings and noise, adjusts its internal hardware and select most suitable parameters for the situation even before the image is captured. In this way, the quality of image will be constant. However, in a practical scenario, some of the images might be affected but there would not be in depth data loss hence data recovery will be possible.

5. Objectives

- To study and determine exact parameters which can be manipulated to produce high quality images.
- To integrate different models of the environment for an efficient solution.
- To exploit the internal architecture of the image sensor.
- To develop image pre-processing IP cores and implement a solution that will integrate all the cores.
- To test the software-defined camera for outdoor surveillance applications under various environmental conditions.

6. Methodology and Equipment/Tools

I. Methodology

Our project's aim is to make camera smart enough, such that it produce high quality images in varying environment. The methodology we will use to achieve our target comprise of:

A. Environmental Profiling:

Conventional cameras will be used for capturing images in different environments at regular intervals. These images help in determining the most varying parameters.

B. Image Sensor Integration:

PCAM 5C module will be connected to the FPGA board and a set of open-source Vivado IP cores will be used for its configuration in software.

C. Image Processing Cores using Vivado HLS:

Vivado High Level Synthesis will be used for generation of different soft IP cores.

II. Equipment/Tools

- FPGA Board (Zybo Z7-10/Zybo Z7-20)
- Image Sensor (PCAM 5C)
- Xilinx Vivado Design Suite
- Xilinx Software Development Kit (SDK)
- Xilinx Vivado High-Level Synthesis (HLS)
- Teraterm

7. Key Milestones and Deliverables

No.	Elapsed time (in months) from start of the project	Milestone	Deliverables
1.	1 month	Literature review	
2.	2 month	Explore Hardware	
3.	3 month	Vivado design suite	
4.	3.5 month	Xilinx SDK	
5.	4 month	Vivado HLS	
6.	4.5 month	Teraterm	
7.	5 month	Image Sensor Integration	Prototype
8.	6 months	IP cores designing	
9.	8 month	Application testing	Final product
10.	9 month	Documentation	Report and Research paper

8. Expected Outcome

A remotely deployable software-defined camera solution that can adjust to different conditions, producing high-quality output video stream with minimal processing time, thus lessening the load of image pre-processing on server-side applications.

9. Direct Customers / Beneficiaries of the Project

Any environment varying image processing application e.g. outdoor surveillance applications.

12. Project Schedule / Milestone Chart



13. Project Approval Certificate

Recommendation of FYP Coordinator

Signature: _____

Approval by the Chairman

Signature: _____

REFERENCES:

- [1] M. Yildirim and A. Çinar, "Simultaneously Realization of Image Enhancement Techniques on Real-Time Fpga," 2019 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, 2019, pp. 1-6
- [2] D. Tsiktisiris, D. Ziouzos and M. Dasygenis, "HLS Accelerated Noise Reduction Approach Using Image Stacking on Xilinx PYNQ," 2019 8th International Conference on Modern Circuits and Systems Technologies (MOCASST), Thessaloniki, Greece, 2019, pp. 1-4, doi: 10.1109/MOCASST.2019.8741574
- [3] Ahmed, E., Ahmed, A., Yaqoob, I., Shuja, J., Gani, A., Imran, M. and Shoaib, M. (2017). Bringing Computation Closer toward the User Network: Is Edge Computing the Solution?. IEEE Communications Magazine, 55(11), pp.138-144.
- [4] H. Kavalionak, C. Gennaro, G. Amato, C. Vairo, C. Perciante, C. Meghini and F. Falchi, "Distributed video surveillance using smart cameras," Journal of Grid Computing, 17(1), 2019
- [5] M. S. AZZAZ, A. MAALI, R. KAIBOU, I. KAKOUCHE, M. SAAD and H. HAMIL, "FPGA HW/SW Codesign Approach for Real-time Image Processing Using HLS," 020 1st International Conference on Communications, Control Systems and Signal Processing (CCSSP), EL OUED, Algeria, 2020, pp. 169-174, doi: 10.1109/CCSSP49278.2020.9151686.
- [6] J. Cong, B. Liu, S. Neuendorffer, J. Noguera, K. Vissers and Z. Zhang, "High-Level Synthesis for FPGAs: From Prototyping to Deployment," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 30, no. 4, pp. 473-491, April 2011, doi: 10.1109/TCAD.2011.2110592.
- [7] S. Lahti, P. Sjövall, J. Vanne and T. D. Hämäläinen, "Are We There Yet? A Study on the State of High-Level Synthesis," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 38, no. 5, pp. 898-911, May 2019, doi: 10.1109/TCAD.2018.2834439.
- [8] O'Loughlin, Declan & Coffey, A. & Callaly, F. & Lyons, D. & Morgan, F.. (2014). Xilinx Vivado High Level Synthesis: Case studies. 352-356