

**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**

# Project Identification

* 1. **Reference Number** (for office use only)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **C** | **S** | **-** | **1** | **8** |  |  |

* 1. **Project Title**

# Software-defined Camera for Outdoor Surveillance Applications

* 1. **Project Internal Advisor**

|  |  |
| --- | --- |
| Name | Fakhra Aftab |
| Designation | Lecturer |

* 1. **Project Internal Co-Advisor**

|  |  |
| --- | --- |
| Name | Dr Majda Kazmi |
| Designation |  |

* 1. **Project External Advisor**

|  |  |  |
| --- | --- | --- |
| Name | Arshad | |
| Designation |  | |
| Organization |  | |
| Mobile # |  | Email |

# Student Team

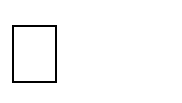
|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Roll No.** | **Name** | **Email** |
| 1. | CS-18133 | Soniya Shafi |  |
| 2. | CS-18118 | Misha Akram | mishaakrambaig@gmail.com |
| 3. | CS-18123 | Iqra Irfan | syedaiqrairfan950@gmail.com |
| ~~4.~~ | CS-18110 | Hoor Soomro |  |

* 1. **Sponsoring Organization** (if any)
  2. **Keywords**

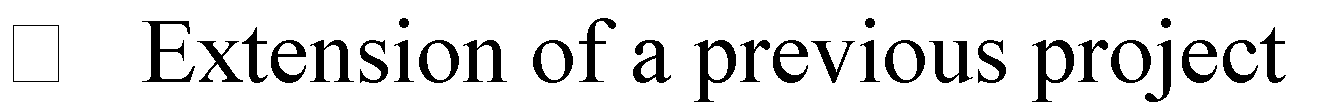
()

* 1. **Project Idea**

Modification to a previous project



New



# ABSTRACT

# Image processing applications are widely used nowadays which require a lot of processing which is a tedious task in order to tackle this problem a solution is devised which will adjust the parameters using environment profiling to capture such images that need little or no further pre-processing to generate accurate images.

# 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.

# 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. Tsiktsiris, D. Ziouzios and M. Dasygenis, "HLS Accelerated Noise Reduction Approach Using Image Stacking on Xilinx PYNQ," 2019 8th International Conference on Modern Circuits and Systems Technologies (MOCAST), Thessaloniki, Greece, 2019, pp. 1-4, doi: 10.1109/MOCAST.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

# 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.

# Objectives

# • Studying and determining exact parameters which can be manipulated to produce high quality images.

# • Use of gathered data to profile the environment.

# • Integrating different models of the environment for an efficient solution.

# • Exploiting the internal architecture of the image sensor.

# • Develop image pre-processing IP cores and implement a solution that will integrate all the cores.

# • Test the software-defined camera for outdoor surveillance applications under various environmental conditions.

# Methodology and Equipment/Tools

# Methodology

# The Xilinx Vivado High-Level Synthesis (HLS) tool will be used for the generation of multiple IP cores using software languages such as C, C++ and system C to handle the lack of the environment (fog, dust, humidity, low light, night etc.). This hardware-software co-design approach will accelerate the time needed to make new IPs and test them. The image sensor will be used to examine the lacking parameters of environment and adjust them accordingly by enabling IP cores before the capture of images.

# 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

1. **Key Milestones and Deliverables**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Elapsed time (in months) from start of the project** | **Milestone** | **Deliverables** |
| 1. |  | Literature review | Proposal |
| 2. |  | Exploring hardware |  |
| 3. |  | Learning tools |  |
| 4. |  | Integrate image sensor | Prototype |
| 5. |  | Testing | Final prototype |
| 6. |  | Documentation | Report and Research paper |

# Expected Outcome

1. **Direct Customers / Beneficiaries of the Project**

# Consent of Advisors

**Consent of the Internal Advisor** Signature:

**Consent of the Co-Internal Advisor** Signature:

**Consent of the External Advisor (if any)** Signature:

# Reviewers Committee’s Comments

1. **Project Schedule / Milestone Chart**

Project schedule using MS-Project (or similar tools) with all tasks, deliverables, milestones, clearly indicated.

|  |  |  |
| --- | --- | --- |
| **13.** | **Project Approval Certificate** |  |
|  | **Recommendation of FYP Coordinator**  **Approval by the Chairman** | Signature:  Signature: |