Image/Video Processing using PYTHON 3 on PYNQ Architecture

Submitted by

Patel Monal Rakeshbhai

Department of E&C Engineering

Parul Institute of Engineering & Technology

Parul University

Research Guide

Dr.Arvind R. Yadav

Ph.D (IIT Roorkee) Image Processing

Associate Dean of Department of E&C Engineering

Parul Institute of Engineering & Technology

Parul University

Signature of the Ph.D Scholar **Research Guide**

H.O.D (Head of Institute) **Dean (DS& R)**

Content of the Pre Ph.D Synopsis

Chapter:1 Introduction 3

Chapter:2 Review of the literature 4

2.1 Crowd Detection and Management using Cascade classifier on ARMv8 and OpenCV-Python 4

2.2 Python Based Image Processing 4

2.3 Detection of moving objects through color thresholding 5

Chapter:3 Scope and statement of the problem 6

Chapter:4 Objective of the study 7

4.1 PYNQ Board 7

4.2 OpenCV and OpenCL 8

Chapter:5 Hypotheses or Research questions 9

Chapter:6 Methodology, tools and techniques 10

6.1 jupyter 10

6.2 SDSoC 11

6.3 SDAccel 12

6.4 OpenVINO Tool Kit 12

Chapter:7 Schedule of the proposed research work 13

References/ Bibliography 14

Chapter: 1 Introduction

Aim of study is the use of hardware resources to accelerate software processes with focus on **video processing**. All elements on how the project is realized are discussed in this report and can be used as beginner guide for new PYNQ users. All HSL, bit, Tcl and Python files are available on github: https://github.com/MonalPatel

This proposal makes use of the PYNQ-Z1 board. This board is the hardware platform for the PYNQ open-source framework. This includes ARM A9 CPUs where the following software runs:

• Linux

• Python

• Jupyter notebook

• Hardware libraries and API for the FPGA

These are used to create a user-friendly and customizable video processing system. Hardware libraries are the programmable logic circuits and are called **overlays**. These are like software libraries. The programmer can select which one matches their application the best. The advantage of using these overlays is that once an overlay is build, it can be reused in other applications.

Setup

For the setup, a laptop is used to generate a 1080p HDMI video signal. This video signal is connected to the PYNQ board in HDMI IN. Here the signal is processed and send back out through the HDMI OUT connector. This connector is connected with a HD 1080p monitor.

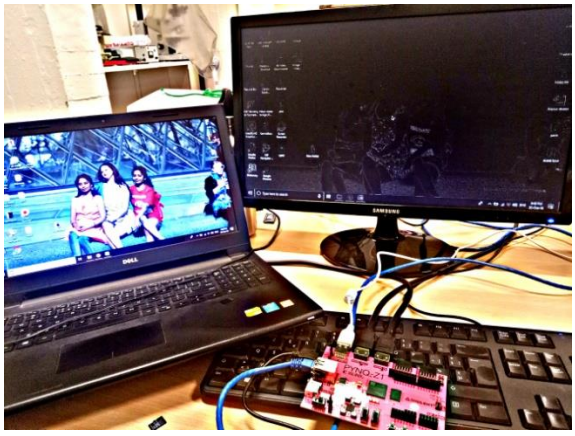

Figure: 1 Setup with PYNQ Board

Chapter: 2 Review of Literature

A series of papers studies is proposed to work on PYNQ board. Specific aims and associated hypotheses of these studies are detailed below

2.1 Crowd Detection and Management using Cascade classifier on ARMv8 and OpenCV-Python [1]

In this paper, the propose is to find a method to manage the crowd by keeping record of the amount of people in the frame. Here the aim is to develop a system using board that consists of ARMv8 CPU which detects the human tops and provide the result with the count of humans in the region using OpenCV-Python. A Haar cascade classifier is trained for human head detection. Human tracking is achieved by indicating the direction of movements of the person. The results of the analysis will be helpful in managing the crowd in any area with high density of crowds.

2.2 Python Based Image Processing [2]

The Python programming language excels at integrated tasks. It is widely used as a high level, free and open source language which is remarkably dynamic, interpreted, scripting and multiparadigm. It also supports object oriented programming features and can be used as a general purpose programming language. Python is easier to learn and has simpler syntax as compared to C, C++ and Java. Python is equally famous for desktop based applications. Python has various and extensive support for libraries such as pip, pillow, Networkx, matplotlib, numpy ,pylab, etc. The fields where Python really shines in are data science and machine learning, numeric, symbolic computations. Also, it is used in other fields like Image processing, Games, Web developments and Big Data Analytics. Python is used by You tube, Google, NASA, Walt Disney, Blender, Cinema 4D, Crystal Space and many more. Image processing with Python is a very efficient and effective process for carrying out operations such as analyzing the digitization of the images to extract the required information. Several operations such as improving the quality, enhancing, zooming, blurring, inverting the image, writing text on the images, greyscale, performing image restoration, recovering, etc. is possible with Python. In this paper, various operations on the image has been performed in Python with the use of functions so that it becomes easy for a person to understand the concepts of Python and image

processing very well. This will be useful for solving the real world tasks and processes in a very effective manner

2.3 Detection of moving objects through color thresholding [3]

In image processing area and segmentation algorithms based on thresholding, the intensity of the image (grayscale) is usually obtained to differentiate between the regions of the objects and the background. The segmentation depending on the threshold works well when the image has a high intensity in the contrast, this characteristic is key feature to get a good classification of the pixels. This paper explains theoretical concepts to identify objects by terms of the color (thresholding), this technique was implemented in the development of a game program. Furthermore, the thresholding range for the red, yellow and green colors was found to get a better approach in the object detection. This project uses the python programming language, Pygame graphical interface libraries and the OpenCV library.

Chapter: 3 Scope and statement of the problem

The scope is to learn the PYNQ environment and use it to accelerate software processes in hardware logic. Basic filters are made here. These filters can be used in more complex systems,such as:

• Canny edge detector

• Image thresholding

• Image Eroding and Dilating

• And more if possible

Building Own Overlay

An overlay comprises of two fundamental parts; the PL structure (bitstream) and the project block diagram Tcl file. Overlay configuration is a particular assignment for hardware engineers. This area accepts the reader has some involvement with advanced structure, building Zynq systems, and with the Vivado Design tools.

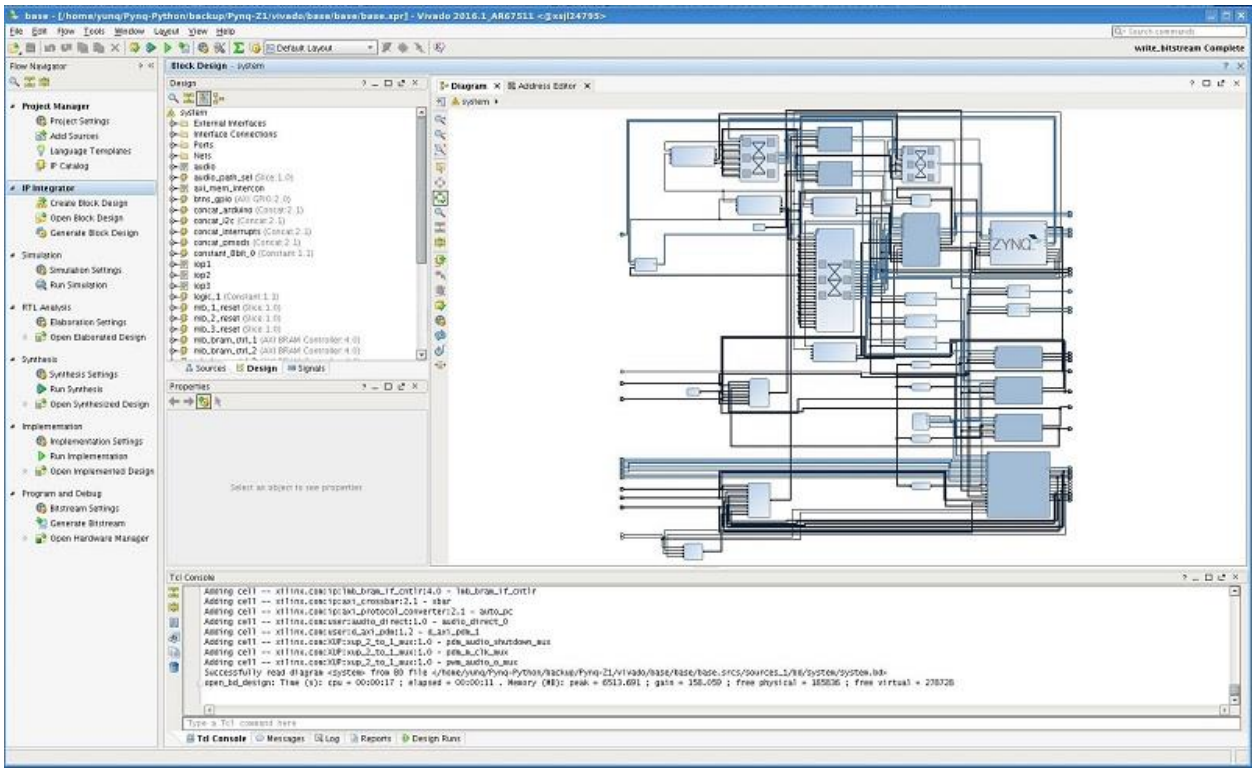

Figure: 2 Create Overlay

Chapter: 4 Objective of study

The main objective of the study is to propose Image and Video process on PYNQ Board using Python and OpenCV and OpenCL and custom video overlay is made to detect an incoming HDMI signal, process it and gives feedback in the form of signal to HDMI output.

There are many overlays in which base overlay permits PYNQ board to utilize Hardware peripherals such as video, audio, GPIOs etc. it connects the IP blocks to the Zynq processing system.

There are many applications which can be made by using this proposal:

- Museum security: to detect the changes by comparing to frames. In this the system will detect missing object automatically by comparing to the previous frame. Even the selection of the previous frame will also automatically done by the system.
- Traffic management: to detect the quantity of vehicles available. By this proposal we can manage the security and quantity of vehicles on the street and even detect specific vehicle too.

4.1 PYNQ: Python Productivity for Zynq

What is PYNQ?

PYNQ is an open-source project from Xilinx that makes it easy to configuration installed systems with Xilinx Zynq Systems on Chips (SoCs). Utilizing the Python language and libraries, originators can utilize the advantages of programmable logic and microprocessors in Zynq to fabricate more skilled and exciting installed systems. PYNQ users would now be able to make superior embedded applications with parallel hardware execution

- high frame-rate video processing
- hardware accelerated algorithms
- real-time signal processing
- high bandwidth IO
- low latency control

Who is PYNQ for?

PYNQ is intended to be used by a wide range of designers and developers including:

- Software developers who want to take advantage of the capabilities of Zynq and programmable hardware without having to use ASIC-style design tools to design hardware.
- System architects who want an easy software interface and framework for rapid prototyping and development of their Zynq design.
- Hardware designers who want their designs to be used by the widest possible audience.

Key technology

Jupyter Notebook is a program based processing environment .Jupyter notebook document can be made that consist of live code, intelligent gadgets, plots, informative content, conditions, pictures and video. A PYNQ empowered Zynq board can be effectively modified in Jupyter Notebook utilizing Python. Utilizing Python, designers can utilize equipment libraries and overlays on the programmable logic. Equipment libraries, or overlays, can accelerate programming running on a Zynq board, and modify the hardware platform and interfaces. The PYNQ image is a bootable Linux image, and consists of pynq Python bundle, and other open-source bundles.

Additional Required Items

- The pynq image and 8 GB SD Card
- Micro-USB cable
- Ethernet cable

4.2 OpenCV and OpenCL :

PYNQ releases ship with the famous OpenCV library pre-introduced. The PYNQ PC vision overlays empower quickening OpenCV segments in Programmable Logic (PL). These overlays uncover a subset Xilinx' xfOpenCV library (a piece of Xilinx' reVISION arrangement) at the Python level, joined with the help for HDMI input/yield (Pynq-Z1 and Pynq-Z2 as it were). Webcam, stream or record based inout/output stays accessible through the pre-introduced SW OpenCV (on all Pynq boards). Currently this package is compatible with PYNQ image v2.3.

SDAccel gives OpenCL attributes to improve the code for information development and kernel execution. The objective of information development streamlining is to amplify the system level information throughput by expanding interface data transmission use and DDR bandwidth utilization. The objective of kernel computation enhancement is to make processing logic that can devour every one of the information when they arrive kernel interfaces.

Chapter: 5 Hypotheses or Research Questions

The following research questions will be investigated and answered:

1) What is & Why PYNQ Board?
2) What is the Jupyter Notebook?
3) What is Github?
4) SDAccel support OpenCV and OpenCL?
5) Which tools will be used?
6) Which overlay used and why?
7) How an overlay created?
8) What are the parameters to create an overlay?
9) What are the limitations of Pynq board?
10) What is PYNQ image and why it is important?

Chapter: 6 Methodology tools and techniques

6.1 Jupyter:

Jupyter notebook is a client server application that enables you to launch Python contents from a website page. These contents keep running on a machine associated with a system, for our situation on the PYNQ map. Jupyter additionally enables you to make files (consequently its name) which are programs that can incorporate content, markdowns, and code to include clarifications all the more clearly or make a tutorial for instance. Jupyter can be introduced on a server or on any PC. Jupyter depends on an IPython kernel for Python 2 and an IPython kernel for Python 3. Jupyter is the application that enables you to program the PYNQ board utilizing Python. all utilization of the board experiences the execution of Python contents, generally by means of Jupyter even if you can likewise utilize the console. In reality the utilization in programming is done by means of Python and hardware coded capacities are additionally called by means of Python. Program the PYNQ board by means of Jupyter with respect to any Python application with the exception of the utilization of overlays. To utilize an overlay you should initially load the overlays library, at that point calling the 'download' capacity of this library you need to load the desired overlay transmitting the way and name of the bitstream that loads the overlay. When the loaded overlay can be characterized python objects that will relate to the IPs of the overlay that you need to utilize. These IPs are controllable by means of the default driver which permits read and write information as bytes or numbers to a given memory address.

The memory deliver to which information can be composed is what is demonstrated after the HLS synthesis. We can modify the drivers to perform more unpredictable capacities in a single line however in the driver that we make we are forced to utilize just the reading function and writing. At the point when the overlay and IPs are loaded we can utilize it as python libraries finally program the application. As of recently the work with Python is done either straightforwardly by means of the interative Python console, or by composing Python programs utilizing a word processor text editor. In any case, there are different approaches to work with Python. Jupyter is an arrangement of devices initially created to make it simpler for researchers

to work with Python and information. It enables to join interactive Python explorations with prewritten programs and even content and equations for documentation.

Jupyter is certifiably not an alternate programming language, it's only an arrangement of PC programs for working with the Python langauage. There are various approaches to introduce Jupyter Notebook.

The Jupyter Notebook is an interative computing environment that empowers users to writer notebook records that include:

- Live code
- Interactive widgets
- Plots
- Narrative text
- Equations
- Images
- Video

The Jupyter Notebook combines three components:

The notebook web application: An interactive web application for writing and running code interactively and authoring notebook documents.

Kernels: Separate processes started by the notebook web application that runs users' code in a given language and returns output back to the notebook web application. The kernel also handles things like computations for interactive widgets, tab completion and introspection.

Notebook documents: Self-contained documents that contain a representation of all content in the notebook web application, including inputs and outputs of the computations, narrative text, equations, images, and rich media representations of objects. Each notebook document has its own kernel.

6.2 SDSoC:

SDSoC is a Xilinx software that looks like Vivado HLS, it can write C code that will be synthesized to turn on the hardware. However SDSoC has more features than HLS because it also allows to write in C the code that will be executed in software, that is to say on the processor. Even though SDSoC is widely used, as part of our application using a map PYNQ and so must be programmed in Python and not in C it is easier to use only HLS because additional SDSoC features are useless. SDSoC does not currently support the

PYNQ board which, unless you create a platform to use SDSoC for PYNQ makes its use for now impossible. Creating such a platform is possible, however, because it is not too long a task when we have the card constraints files (which is the case) and this would allow to many SDSoC users to be able to carry a much larger C application easily on the PYNQ than the way in which we should proceed now.

6.3 SDAccel :

The Xilinx SDAccel Development Environment is part of the SDx Development Toolchain. This toolchain allows you to create FPGA accelerated designs using C/C++, OpenCV, C, or RTL programming languages. You can create these designs in the SDx GUI environment or through a make file flow, building a basic OpenCL based design using the SDx GUI and learning some of the features that enable you to do performance profiling and or optimization

6.4 Openvino toolkit:

The **OpenVINO toolkit** is designed to enable users to fast-track development of high-performance computer vision applications, unleash deep learning inference capabilities across the entire **Intel** silicon portfolio, and provide an unparalleled solution to meet their AI needs.

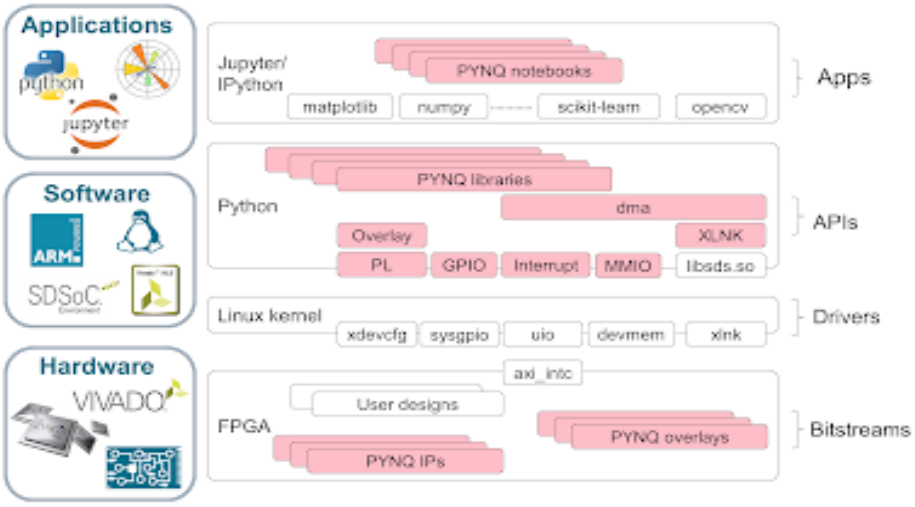

Figure: 3 Hardware /software

Chapter: 7 Schedule of the proposed research work

Sep 2018:

- Literature Survey

Oct 2018:

- PYNQ Board setup
- Jupyter tutorials

  1)HDMI introduction .ipynb
  2)HDMI video pipeline.ipynb
  3)OpenCV face detect.ipynb
  4)OpenCV face detect webcam.ipynb
  5)OpenCV filter hdmi.ipynb
  6)OpenCV filter webcam.ipynb

- GitHub Account made and upload documents

Nov 2018:

- SDAccel , SDSoC,OpenVINO Toolkit

Installation, Setup, Tutorials

Dec 2018:

- Understanding OpenCV and python Libraries
- Performing tutorials

Jan 2018:

- Learning overlay:
  1) to make
  2) parameters important for an overlay
  3) learning tools used to make overlay
- Creat own Overlay using SDSoC and SDAccel

Feb 2018:

- Hardware Implement

Mar 2018:

- Making content for OPEN HARDWARE CONTEST 2019

  www.openhw.eu0

Reference / Bibliography

Papers

1) Crowd Detection and Management using Cascade classifier on ARMv8 and OpenCV-Python
2) Python Based Image Processing
3) Detection of moving objects through color thresholding
4) Hot & Spicy:Improving productivity with Python and HLS for FPGA
5) Real time object detection & tracking system with rotating camera
6) Evaluating Rapid Application Development with Python for Heterogeneous Processor-based FPGAs.

Websites

1) **http://www.pynq.io/**
2) **https://www.xilinx.com/support/university.html**
3) **https://github.com/**
4) **http://jupyter.org/**
5) **https://www.xilinx.com/products/design-tools/software-zone/sdaccel.html**