Senior Project – Computer Engineering – 2020

Sign Detection System for Real Time Applications

Brock Harris, Will Christensen

Advisor – Prof. Cherrice Traver

Introduction

Image processing can be a significant undertaking for traditional processor to compute. Because of this, some image processing applications take advantage of a hardware software codesign system. We are implementing this the combination of an FPGA and an HPS system.

System Overview

Algorithm: We chose to use the Histogram of Oriented Gradients (HOG) feature detector, implemented on an FPGA, and a Support Vector Machine (SVM) implemented on an ARM hard processor.

We will be using a HW/SW System on a Chip (SoC). This board allows us to implement software on the onboard ARM chip and also hardware on the FPGA.

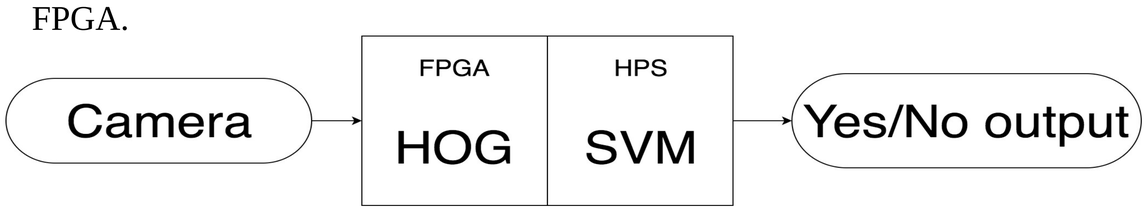

Figure 1: Top level design overview

Hardware-Software Co-design

Quartus: Software used to design programs for the FPGA section of the SoC. We have implemented a simple project on the FPGA to prove that we can use the FPGA.
Qsys: Software to allow for communication between the FPGA and the HPS. This allows us to assign data to buses and transfer the data across clock domains. We plan to implement a test project to show the connection by the end of the term.
Linux: Embedded operating system running on the HPS Arm processor. Allows us to run C programs and communicate with FPGA.

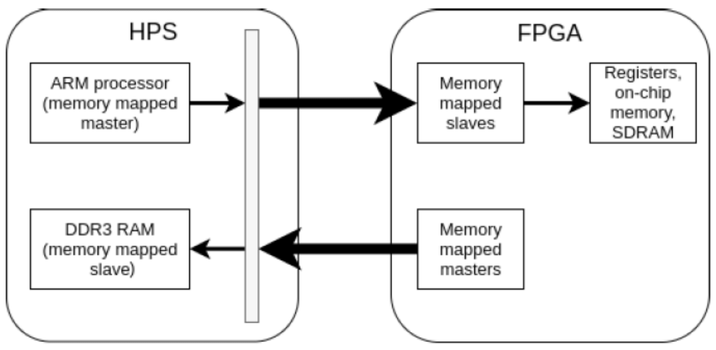

Figure 2: FPGA SoC connection

Algorithm Overview

Preprocessing:
- Get the dominate color vector for each pixel
- Get magnitude and orientation for gradients over the image

HOG:
- Collect the gradients together for 8x8 pixel cells
- Vector normalization
- Compress the vectors to save space

SVM:
- Classification of images based on training

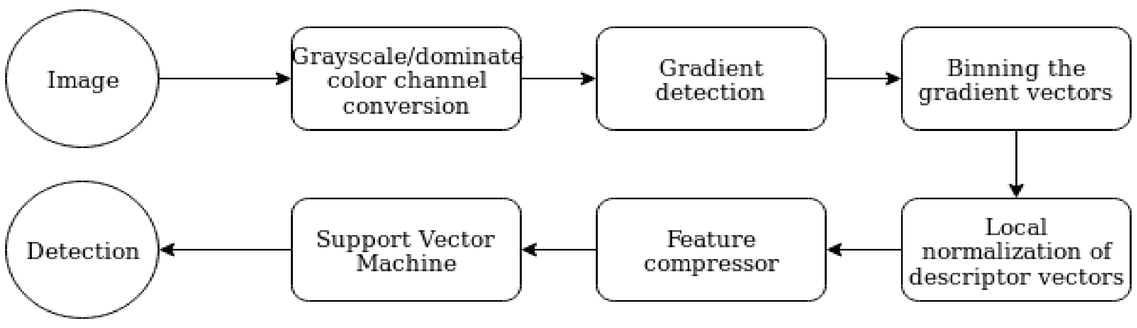

Figure 3: Flow diagram for proposed algorithm

Current Implementation

Hardware
- Reading data from the Camera: We get images streaming from the camera convert them from Bayer pattern to Gray scale or RGB.
- Computing Sobel Filter: Convert RGB Image into edge detected image and display that on the VGA Monitor.
- Computing Gradient magnitude values: We can use these to find the dominate color gradient color channel in the image.

Hardware Software Communication
- Memory mapped pointer implementation in C: Implemented a basic bus comm. between FPGA and HPS.
- First In First Out (FIFO) memory implementation: Using a FIFO memory component to alleviate synchronization issues between the FPGA and HPS

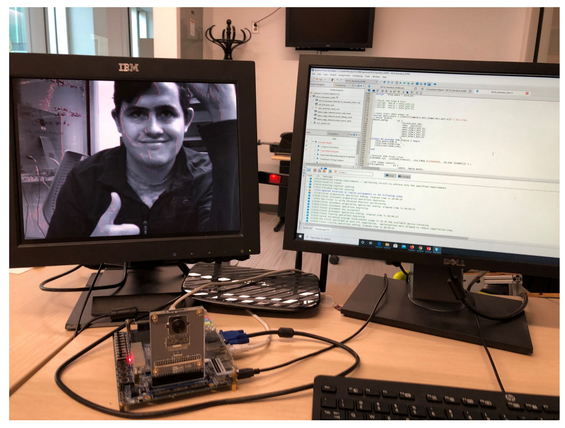

Figure 4: Our setup

Issues

- Changing clock domain: Transferring data between components with two different clock speeds presents challenges that can lead to incorrect data processing.
- Pin assignments: Placing pin in a design is essential and the documentation is not always clear and errors are often vague
- Compilation Errors: Qsys and Quartus warnings can sometimes cause the system to not compile. They do not provide any detail as to what the issue is. This was a major slow down to
- Communicating with the FIFO: The block used for this has not been properly debugged, it reads values too slowly or incorrectly this caused errors in obtaining real results

Results

Due to unforeseen complexities in the project we were unable to complete the full algorithm. This means that despite making significant progress we did not meet our original design criteria. This left us with a system that:
- Was able to communicate between the custom hardware and software developed
- Separate Hardware components able to compute magnitudes of HOG gradients but unable to find angle.
- A system with unreliable synchronization and unreliable communication between hardware and software.
- Complications in hardware regarding when data is going where.

Future Work

- Finish HOG algorithm implementation: We still need to create the binning and histogram generation components.
- Connecting Hardware components: Connecting the components together so that the data can be transferred between them.
- Implement and Training HPS ready SVM: We have a dataset to train on. We just need to set up the SVM^{light} for the HPS
- Stabilize and debug HW/SW communication: While we have implemented a communication system that will work for our purposes we need to verify that it will work in all circumstances.
- Fix bugs HW/SW synchronization: In current system with HW/SW communication issues have arose where data can be skipped or read multiple times

Acknowledgments

Dr. Tony Frangieh[#] Prof. Cherrice Traver[+] Prof. Shane Cotter[+] Prof. Matt Anderson[*] Prof. John Spinelli[+]

[*] Computer Science Department [+] ECBE Department [#] Schneider Electric