MCGILL UNIVERSITY

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

ECSE 457 - FINAL REPORT

Research & Development of a Real-Time Object Tracking System

Authors:

Benjamin BROWN
benjamin.brown2@mail.mcgill.ca
260450182

Taylor DOTSIKAS
taylor.dotsikas@mail.mcgill.ca
260457719

Supervisors:

Warren GROSS, PROF.

Arash ARDAKANI

Sponsored by:

ANALOG DEVICES

Abstract

Acknowledgments

We would like to thank Professor Warren Gross and Arash Ardakani for overseeing this project and providing advice, insight, and direction over the course of the year. We would also like to thank Analog Devices for providing us with the Altera DE2 board needed for hardware implementation, and in particular Leah Magaldi for being our main point of contact. Finally, the example TV decoder Verilog code provided by Terasic was a major asset for hardware implementation, we would like to thank them for making it open and available to developers like ourselves.

Contents

1 Abbreviations & Notation

FPGA - Field Programmable Gate Array

VGA - Video Graphics Array

2 Introduction

Hello World

3 Background

This section contains the prerequisite information regarding video tracking needed to understand the system architecture and design. For background information regarding the basics of video processing, Kalman filtering, fixed-point representation, and optical flow, please see [**?**].

3.1 Determining Position

The algorithm implemented in software, and presented in [**?**], for determining the (x, y) position of an object in the delta frame used a rastor scan technique to determine the leftmost, rightmost, top, and bottom pixels. By intersecting two lines formed between these points, the center of the object can be estimated. It was found that despite the success of this algorithm in software, it would not be conducive to hardware implementation.

3.2 Video Pipeline

Describe a video pipeline and common components.

3.3 The VGA Interface

Discuss the timing signals

4 Requirements

5 Design

5.1 Generating VGA Output

Make sure mention the design references for this.

5.2 Simple Video Pipeline

This is where you should cite [**?**] and talk about the video input wrapper module.

5.3 Storing the Base Frame

This is where you should cite [**?**] and talk about the SRAM.

5.4 RGB to Grayscale Conversion

Simple discussion here.

5.5 Modified Video Pipeline

Talk about moving the RGB conversion further upstream prior to the SDRAM such that we can work in RGB.

5.5.1 Delta Frame Generation

Discuss the poor man's absolute value implemented and the saturation filter.

5.5.2 Moving Average Filter

Show the two pictures of the before/after moving average filter here.

5.6 Measuring Object Position

Make sure to mention it's using the VGA module counters.

5.7 Kalman Filter

Lots of meat here.

6 Future Work

The system could be significantly improved by using a measurement algorithm that has fewer requirements and assumptions regarding the environment. Variations in lighting and changes in the background cause the delta frame generation to break, and in a real-time situation we found it difficult to ensure that all the requirements were met.

7 Impact on Society

8 Allocation of Work

9 Conclusion

The final hardware implementation was a success, and the system can display a red dot that shows the (x, y) measurement of the object in the frame as well as green dot showing the improved (x', y') position the Kalman filter produces. Due to the underlying principles of the delta frame generation, the system is an object tracking algorithm and not a motion tracking algorithm. However, the Kalman filter theoretical model assumes constant velocity.This is why when the object stops moving, the green dot slowly converges to the red one. This shows the system favoring the measurements over the model to reduce error, and demonstrates the Kalman filter functioning properly. When the object is moving with constant velocity, the Kalman filter provides a smoother, improved output.

References

[1] F. Roth, “Using low cost FPGAs for realtime video processing”, M.S. thesis, Faculty of Informatics, Masaryk University, 2011.

[2] A. Saeed et al., “FPGA based Real-time Target Tracking on a Mobile Platform,” in 2010 International Conference on Computational Intelligence and Communication Networks, 2010, pp. 560-564.

[3] ”Video and Image Processing Design Using FPGAs.” Altera. 2007. January 2014. http://www.altera.com/literature/wp/wp-video0306.pdf

[4] E. Trucco and A. Verri, “Chapter 8 - Motion,” in Introductory Techniques for 3D Computer, pp. 177-219.

[5] S.A. El-Azim et al., “An Efficient Object Tracking Technique Using Block-Matching Algorithm”, in Nineteenth National Radio Science Conference, Alexandria, 2002, pp. 427 - 433.

[6] Caner et al., “An Adaptive Filtering Framework For Image Registration”, IEEE Trans. Acoustics, Speech, and Signal Processing, vol. 2, no. 2, 885-888. March, 2005.

[7] Yin et al. *Performance Evaluation of Object Tracking Algorithm* [Online]. Available: http://dircweb.kingston.ac.uk/

[8] G. Shrikanth, K. Subramanian, “Implementation of FPGA-Based object tracking algorithm,” Electronics and Communication Engineering Sri Venkateswara College of Engineering, 2008.

[9] E. Pizzini, D. Thomas, “FPGA Based Kalman Filter,” Worcester Polytechnic Institute, 2012.

[10] M. Shabany. (2011, December 27). *Floating-point to Fixed-point Conversion* [Online]. Available: http://ee.sharif.edu/ digitalvlsi/Docs/Fixed-Point.pdf

[11] N. Devillard. (1998, July). *Fast median search: an ANSI C implementation* [Online]. Available: http://ndevilla.free.fr/median/median.pdf

[12] D. Kohanbash. (2014, January 30). *Kalman Filtering - A Practical Implementation Guide (with code!)* [Online]. Available: http://robotsforroboticists.com/kalman-filtering/

[13] Recommendation ITU-R BT.656-5 (12/2007). “Interface for digital component video signals in 525-line and 625-line television systems operating at the 4:2:2 level of Recommendation ITU-R BT.601”.

[14] ISSI Datasheet: “1M x 16 High-Speed Asynchronous CMOS Static RAM with 3.3V Supply”.

[15] B. Brown & T. Dotsikas, “A Real-Time Object Tracking System”, ECSE 456 Final Report, unpublished.

Appendices