A Platform-Based SoC Design for Real-Time Stereo Vision

Jongsu Yi*, Jaehwa Park, and JunSeong Kim***

Abstract—**A stereo vision is able to build three-dimensional maps of its environment. It can provide much more complete information than a 2D image based vision but has to process, at least, that much more data. In the past decade, real-time stereo has become a reality. Some solutions are based on reconfigurable hardware and others rely on specialized hardware. However, they are designed for their own specific applications and are difficult to extend their functionalities. This paper describes a vision system based on a System on a Chip (SoC) platform. A real-time stereo image correlator is implemented using Sum of Absolute Difference (SAD) algorithm and is integrated into the vision system using AMBA bus protocol. Since the system is designed on a pre-verified platform it can be easily extended in its functionality increasing design productivity. Simulation results show that the vision system is suitable for various real-time applications**

Index Terms—**Stereo vision, SAD correlator, platform based design, SoCBase, real-time system**

I. INTRODUCTION

Vision sensors provide rich sources of information providing timeliness and affordable services in applications including robots, factory automations, intelligent vehicles, and home networks. As a passive system, they are much less sensitive to environmental interference and can provide rich sources of information for scene recognition, motion detection, object tracking, surveillance, and so on. Vision sensors, however, generate high bandwidth data due to the nature of images. Standard PCs are commonly used for image analysis but processing even small low resolution images takes more than a second in software. This is well below the frame rates obtainable with commodity cameras, which can provide 30 or more images per second, and may be far too slow to provide services in real-time applications. Sensing multiple images simultaneously and processing them in real time, even in a low resolution, would be a challenging task.

Stereo vision, which is based on two or more images taken from different viewpoints, is able to build three-dimensional maps of its environment [1, 4]. It can provide much more complete information than two-dimensional image based vision and is actively worked in many application areas including intelligent robots, autonomous vehicles, smart surveillance and security [11, 14, 15]. In the past decade, real-time stereo has become a reality. Some solutions are based on reconfigurable hardware and others rely on specialized hardware. For example, a stereo system for household mobile robot using Xilinx XC2V3000 FPGA runs at 60 fps for 640×480 images with disparity of 128 [7]. The PARTS reconfigurable engine using 16 Xilinx 4025 FPGAs runs 42 fps for 320×240 images with disparity of 24 [3]. The CMU video-rate stereo machine using an array of TMS320C40 DSP processors runs at 30 fps for 200×200 images with disparity of 32 [2]. The stereo system by S. Jin et al. using a Virtex-4 XC4VLX200-10 FPGA runs 60 fps for 640×480 images with disparity of 64 [13]. The

Manuscript received Dec. 28, 2011; revised Jan. 29, 2012.
* School of Electrical and Electronics Engineering, Chung-Ang University, Seoul, Korea
** Department of Computer Engineering, Chung-Ang University, Seoul, Korea
E-mail : junkim@cau.ac.kr

FingerMouse using an ASIC runs at 30 fps for 320×240 images with disparity of 47 [8]. Each of them, however, is designed for a specific application and is difficult to extend its functionality. A platform-based design alleviates the problem allowing a vision system to be easily modified, thus increasing design productivity [5].

This paper describes an extensible vision system based on the SoCBase, which is a design environment containing essential IPs such as bus, peripheral and memory controllers. We implement a SAD correlator as a peripheral module being integrated into the vision system with additional components such as SAD wrapper, memory controller, TFT-LCD controller and keypad controller. The platform-based design allows the vision system to be easily modified for specific applications.

The remainder of this paper is organized as follows. Section II briefly surveys the SoCBase platform. Section III provides the SAD stereo image matching technique and the implementation of the vision system prototype. Some conclusions and future works are summarized in Section IV.

II. SoCBase Platform

Advances in CMOS technology integrating multi-million transistors in a single chip make it feasible to build complex systems for various applications. As the design complexity of digital systems grows platform-based SoC design methodologies are widely accepted [6, 9, 16]. With a well-defined SoC platform, complex systems can be designed in less effort and time by reusing components IPs. In addition, derivative designs become easy by modifying or adding few components from once established systems. SoCBase is a generic System-on-Chip design platform developed by the Center for SoC Design Technology, Seoul National University, Seoul, Korea [6, 17]. It is an abstraction, that hides details of possible implementation refinements, including a library of reusable IPs, effective design tools and verification flows. Fig. 1 shows the configuration of the SoCBase, which mainly consists of platform hardware and platform software.

The platform software contains operating system, verification program, and applications. Linux and verification purpose OS(VPOS), which is a light-weight OS designed for the SoCBase platform, are provided with device drivers for various component IPs.

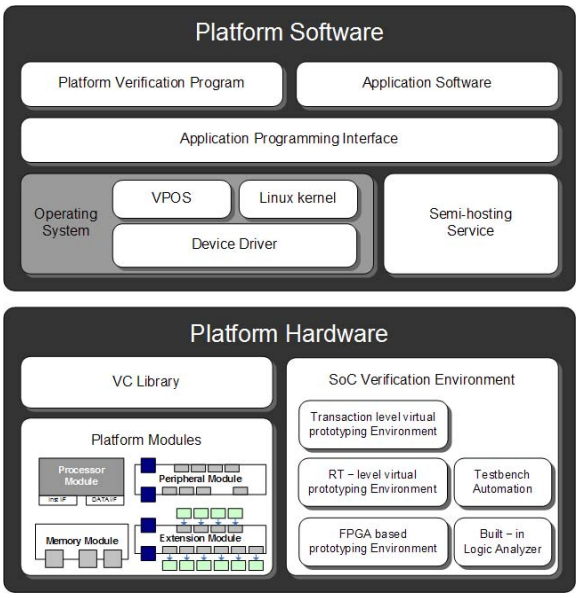

Fig. 1. The configuration of the SoCBase platform.

Verification program is a collection of methods for rapid verification of a system and its components. Both verification program and application software are implemented using platform API.

The platform hardware contains VC (Virtual Component) library, platform modules, and verification environment. The VC library provides more than 30 hardware component IPs including AMBA bus components, memory controllers, and I/O peripherals. Platform modules are subsystems with many component IPs that are commonly used together being integrated in a single module. Based on the AMBA 2.0 compliant bus, platform modules can be combined with each other or with other IPs for building a system. It is the platform modules that allow designers not to consider the detailed modeling of a system increasing the design productivity. In addition, the verification environment reduces design time with the supports of HW/SW co-simulation, testbench automation tools, and a built-in logic analyzer.

SoCBase 1.0 is a base platform targeting low power embedded systems. It is a single processor centric architecture supporting widely-used ARM cores.

III. A Vision System Design

1. Stereo Image Correlation

Stereo vision can be used for extracting three

dimensional structure from two or more images taken from different viewpoints. It has to determine, given a pair of stereo images, which parts in one image correspond to which parts in the other image. Ideally, we want to find all matching pixels of a pair of stereo images. However, the value of a single pixel has too little to determine its correspondence. Instead, sets of neighboring pixels, called *image windows*, are used for practical stereo matching algorithms. That is, given a pair of stereo images, one window is fixed in one image and another window is moving in the other image. By comparing them image correlation is measured and the matching window, that maximizes the similarity criterion, is determined. Common metrics are Normalized Cross-Correlation (NCC), Sum of Squared Differences(SSD), Sum of Absolute Differences(SAD), Census, and Rank algorithms [4, 10, 12].

We use the SAD correlation algorithm because the matching steps can be calculated all in parallel. In addition, regular structures and linear data flow make the SAD algorithm a good candidate for hardware implementations. The SAD function is defined to be

$$C(x,y,\delta)=\sum_{y=0}^{wh-1}\sum_{x=0}^{ww-1} I_R(x,y)-I_L(x+\delta,y) \quad (1)$$

where $I_R(x,y)$ and $I_L(x,y)$ represent coordinates in pixel on the right image and the left image, respectively. The δ represents disparity number ranging between 0 and the maximum disparity value Δ. The *ww* and *wh* represent the window width and height. The SAD function $C(x,y,\delta)$ is evaluated for all possible values of the disparity, δ, and the minimum is chosen. That is, the criterion for the best match in the SAD algorithm is minimization of the sum of the absolute differences of corresponding windows.

We implement a SAD correlator to find the best match among image windows. Fig. 2 shows a block diagram of the SAD correlator. It is implemented with only adders and comparators and we use the topology of tree to provide the performance of O(log *wh*+*ww*+Δ). The SAD correlator consists of three modules: *pixel shift register*(PSR), *disparity calculator*(DC), and *minimum calculator*(MC) modules:

- The PSR module provides sufficient pixel streams from both cameras to the DC module. It is organized as a type of one-dimensional array with the size of "*scan line length* × (*wh*-1) + Δ". In order to exploit the parallelism in SAD algorithm, a single window from the left-PSR and (Δ+1) windows from the right-PSR submodules are transmitted to the DC module simultaneously.
- The DC module, which is further divided into *column-DC*, *shift-buffer*, and *window-DC* submodules, executes the SAD function $C(x,y,\delta)$. By separating column-SAD calculations from the SAD function we can eliminate redundant operations among neighboring windows. First, the column-DC submodule calculates column-SAD results for a pair of columns from each window. Next, the window-DC submodule calculates SAD results for a pair of window with the corresponding column-SAD results. The shift-buffer submodule, which is organized as a two-dimensional array of the size "(Δ+1) × *ww*", is placed in between the two

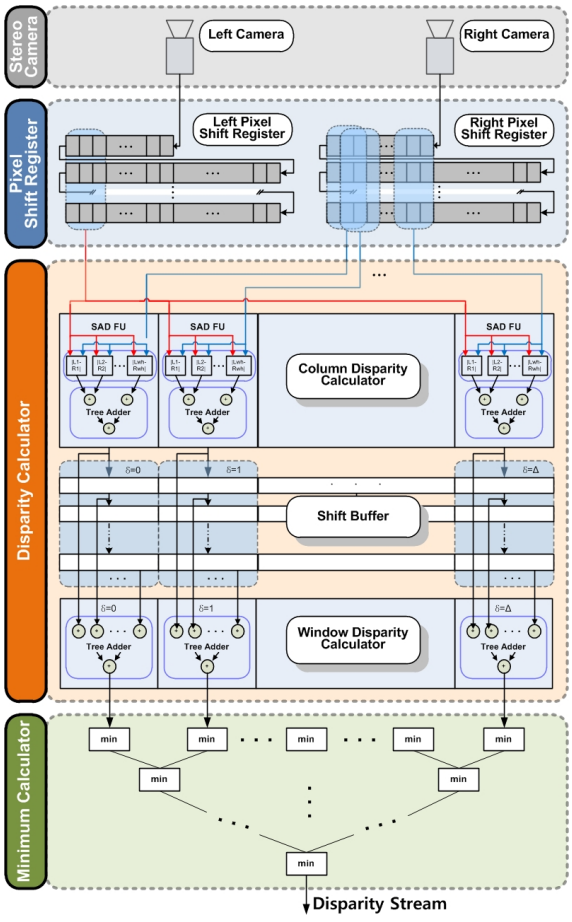

Fig. 2. A block diagram of the SAD correlator.

submodules. It stores the column-SAD results and selectively provides them to the window-DC submodule.

- The MC module determines the best fit pair of windows among the (Δ+1) SAD results from the DC module.

2. A Vision System Prototype

We implement a stereo vision system using the Altera Excalibur processor supporting SoCBase 1.0 platform. Fig. 3 shows the block diagram of the vision system. Three platform modules are utilized: the processor module embeds an ARM922T core and the memory module contains memory controllers and the peripheral module consists of low speed devices including a stereo image correlator, a TFT-LCD controller, a keypad controller. Using two AMBA advanced high performance buses (AHBs) it provides efficient communications among the three platform modules. There is a direct AHB connection between the processor and memory modules for fast data processing. The peripheral module groups all low speed devices under the other AHB bus. Any devices in the peripheral module go through AHB bridges for communication with the processor module. This allows the SAD correlator to handle data in memory while the processor operates simultaneously in the control plane.

The SAD correlator is implemented in hardware, as shown in Section III.1, considering real-time operations

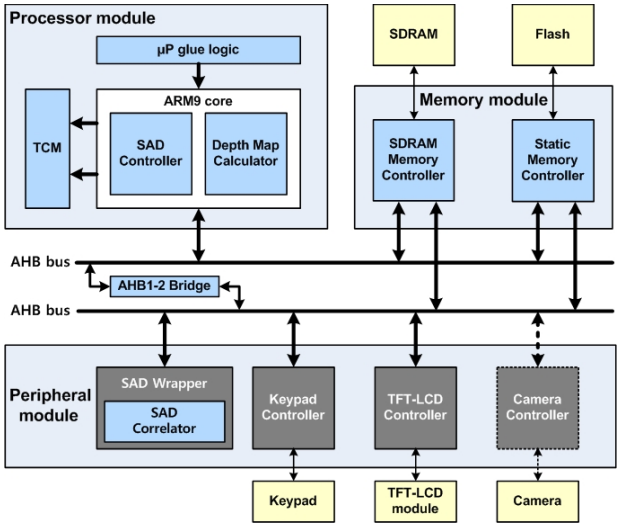

Fig. 3. A stereo vision system prototype using SoCBase platform.

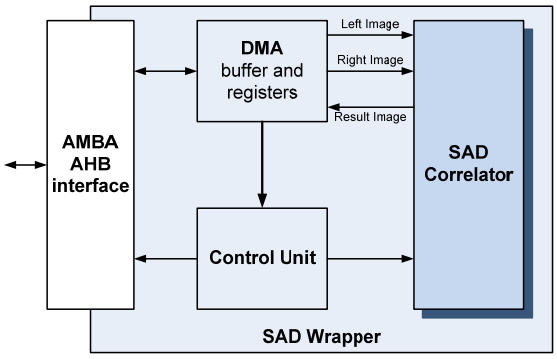

Fig. 4. A block diagram of the SAD wrapper.

since it is the most compute intensive part in the system. The interface of the SAD correlator has to support AMBA AHB protocol so that it can easily integrated into the vision system using SoCBase. Fig. 4 shows a block diagram of the SAD wrapper, which converts the I/O signals of the SAD correlator into AMBA AHB compliant ones and controls data transmissions among IPs in AMBA AHB protocol. It also supports DMA (direct memory access) operations since the SAD correlator accesses memory with a large amount of data.

The initialization, interrupt handling, and the system controls are implemented in software considering flexibility and extensibility. The flow of operations for the vision system is as follows: ⓐ Once the system is triggered by a key stroke after initialization it brings a pair of stereo images from memory (ultimately, they will come from a pair of cameras). This is done by using DMA saving processor's clock cycles for other operations. ⓑ The SAD correlator finds the best match for the stereo images and issues an interrupt so that the disparity results can be saved into memory. ⓒ The processor calculates depth map based on the disparity results and saves it into memory. ⓓ The depth map is displayed on a TFT-LCD device with proper data conversions. For timely handling of interrupt requests priorities are given to the SAD wrapper, TFT-LCD, keypad, and UART in the order because the main operation of the vision system comes from the SAD correlator.

3. Experimental Results

The key parameters determining the size and performance of the vision system are the scan line length

Table 1. The resource usage of the SAD IP with various window sizes

Window size (*ww*×*wh*)	Logic cells	Utilization
5x5	7,806	20.3%
7x7	11,576	30.1%
9x9	18,550	48.3%

Table 2. The resource usage of the peripheral module for the stereo vision system

	Logic cells	LC registers	Memory bits
SAD correlator	18,550	11,362	22,344
TFT-LCD controller	624	369	16,384
Total	19,243	11,739	38,728

(*sl*), the window width (*ww*), the window height (*wh*), and the maximum disparity (Δ). The Table 1 shows the resource usage for the SAD IP, which includes both the SAD correlator and the SAD wrapper with various window sizes. It has been compiled with the target of Excalibur EPCA10F1020C2, which has 38,400 logic cells in its PLD portion. For the table, we set the maximum disparity for Δ=31, with the scan line length of *sl*=240. The Table 2 shows the resource usage for the peripheral module of the stereo vision system, which includes the SAD IP with 9x9 windows and a TFT-LCD controller. From the table we conclude that the system can be easily fit into a single reconfigurable device even with additional hardware IPs for pre- or post-processing of the vision system.

To evaluate the performance of the vision system we use sets of widely used stereo images from the Middlebury Stereo Vision Page as test inputs [18]. Fig. 5 shows a sample of resulting depth maps of the vision system. The Tsukuba image consisting of 320×240 pixels is used as the input. Simulation with the Tsukuba image shows the total execution time of 8,070 ms. It means that the vision system can process about 123 images per second. Though we did not recompile the vision system for higher resolution images we can easily estimate its processing rate. For example, images of 640×480, whose number of pixels is four times to that of a 320×240 image, would take about 30 frames per second processing rate.

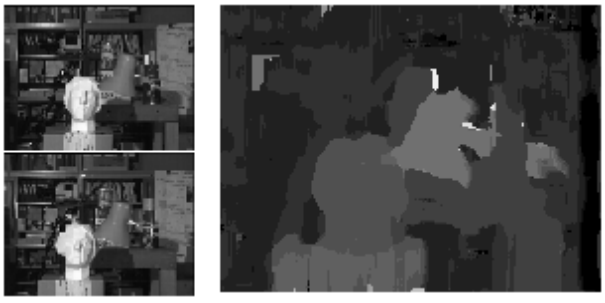

Fig. 5. A sample depth map result using Tsukuba image.

IV. CONCLUSIONS

Vision provides plentiful information and can be the most powerful means to deduce an environmental situation for both human and machines. However, processing even a small image takes seconds in software, which is far below the demands in real-time applications, and hardware accelerations are necessary. In this paper, we describe an extensible vision system providing three dimensional information. A simple SAD correlator is implemented as a peripheral module based on the SoCBase platform. In simulation, we show its functionality for real-time applications. Since it is integrated with a predefined platform the vision system can be easily extended its capability for various applications. For instance, addition of post- and pre-processing modules could be implemented in either hardware or software without additional difficulty for integration with the vision system.

ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (grant number 2012-0001454).

REFERENCES

[1] S. T. Barnard, M. A. Fischler, "Computational Stereo," *ACM Computing Surveys (CSUR) archive*, Vol.14, pp.553-572, 1982.

[2] T. Kanade, A. Yoshida, K. Oda, H. Kano, M. Tanaka, "A stereo machine for video-rate dense depth mapping and its new applications," *Computer Vision and Pattern Recognition*, pp.196-202, 1996.

[3] J. Woodfill, B. V. Herzen, "Real-time stereo vision

on the PARTS reconfigurable computer," *IEEE symposium on FPGAs for Custom Computing Machines*, pp.242 - 250, 1997.

[4] M. Z. Brown, D. Burschka, G. D. Hager, "Advances in computational stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.25(8), pp.993-1008, Aug. 2003.

[5] A. Sangiovanni-Vincentelli, L. Carloni, F. D. Bernardinis, M. Sgroi, "Benefits and Challenges for Platform-Based Design," *Proceedings of Design Automation Conference*, pp.409-414, 2004.

[6] S. Park, M. Kim, S. I. Chae, "SoCBase - An integrated solution for platform based design," *International SoC Design Conference*, pp.329-332, 2004.

[7] D. Han, D. H. Hwang, "A novel stereo matching method for wide disparity range detection," *Lecture Notes on Computer Science*, Vol.3656, pp.643-650, 2005.

[8] P. de la Hamette, G. Troster, "FingerMouse-Architecture of an ASIC-based Mobile Stereovision Smart Camera," *IEEE International Symposium on Wearable Computers*, 2006.

[9] S. Park, S. Yoon, S. I. Chae, "Reusable component IP design using refinement-based design environment," *Asia and South Pacific Conference on Design Automation*, 2006.

[10] H. Hirschmuller, D. Scharstein, "Evaluation of Cost Functions for Stereo Matching," *Computer Vision and Pattern Recognition*, 2007.

[11] K. Konolige, m. Agrawal, R.C. Bolles, C. Cowan, M. Fischler, B. Gerkey, "Outdoor Mapping and Navigation Using Stereo Vision," *Experimental Robotics, Springer Tracts in Advanced Robotics*, Vol.39, pp.179-190, 2008.

[12] M. Humenberger, C. Zinner, M. Weber, W. Kubinger, M. Vincze, "A fast stereo matching algorithm suitable for embedded real-time systems," *Computer Vision and Image Understanding*, Vol. 114, pp.1180-1202, Nov., 2010.

[13] S. Jin, J. Cho, X. D. Pham, K. M. Lee, S. K. Park, M. Kim, J. W. Jeon, "FPGA Design and Implementation of a Real-Time Stereo Vision System," *IEEE Transactions on Circuits and Systems for Video Technology,* Vol.20(1), pp.15-26, Jan. 2010.

[14] A. Cappalunga, S. Cattani, A. Broggi, M. S. McDaniel, S. Dutta, "Real time 3D terrain elevation mapping using ants Optimization Algorithm and stereo vision," *IEEE Intelligent Vehicles Symposium,* pp.902-909, 2010.

[15] J.-H. Sun, B.-S. Jeon, J.-W. Lim, M.-t. Lim, "Stereo vision based 3D modeling system for mobile robot," *Control Automation and Systems (ICCAS), 2010 International Conference on*, pp.71-75, Oct., 2010.

[16] J. Um, S. Hong, Y.-T. Kim, E. Chung, K.-M. Choi, J.-T. Kong and S.-K. Eo, "ViP: A Practical Approach to Platform-based System Modeling Methodology," *Journal of Semiconductor Technology and Science,* Vol.5, No.2, pp.89-101, Jun., 2005.

[17] Center for SoC Design Technology Page, http://soc.snu.ac.kr/

[18] Middlebury Stereo Vision Page, http://vision.middlebury.edu/ stereo/

Jongsu Yi received a Ph.D. in Electronics Engineering from Chung-Ang University in Seoul, Korea. His research interests include computer architecture, intelligent home service and stereo vision system. He is a member of the IEICE and a member of the IEEK.

Jaehwa Park received a Ph.D. degree in Electrical Engineering from State University of New York at Buffalo, a M.S. and a B.S. both in Electrical Engineering from Hanyang University, Korea. He is currently an Associate Professor in the Department of Computer Engineering at the Chung-Ang University in Seoul. His research interests include pattern recognition and human interface.

JunSeong Kim received a Ph.D. in Electrical Engineering from the University of Minnesota at Minneapolis, a M.S. and a B.S. both in Electronics Engineering from Chung-Ang University in Seoul, Korea. He is currently a Professor of School of Electrical and Electronics Engineering at the Chung-Ang University in Seoul. His main research interests include computer architecture, high-performance computing, parallel processing, and embedded system design.