

영상처리를 위한 SIMT 기반 Image Signal Processor 구현

Implementation of the SIMT based Image Signal Processor for the Image Processing

황 윤 섭*, 전 희 경**, 이 관 호**, 이 광 엽***

Yun-Seop Hwang*, Hee-Kyeong Jeon**, Kwan-ho Lee**, Kwang-yeob Lee***

Abstract

In this paper, we proposed SIMT based Image Signal Processor which can apply various image preprocessing algorithms and allow parallel processing of application programs such as image recognition. Conventional ISP has the hard-wired image enhancement algorithm of which the processing speed is fast, but there was difficult to optimize performance depending on various image processing algorithms. The proposed ISP improved the processing time applying SIMT architecture and processed a variety of image processing algorithms as an instruction based processor. We used Xilinx Virtex-7 board and the processing time compared to cell multicore processor, ARM Cortex-A9, ARM Cortex-A15 was reduced by about 71 percent, 63 percent and 33 percent, respectively.

요 약

본 논문에서는 다양한 영상 전처리 알고리즘들을 적용할 수 있고, 영상 인식과 같이 ISP 응용 프로그램을 병렬로 처리 가능한 SIMT(Single Instruction Multiple Threads) 기반 Image Signal Processor를 제안하였다. 기존의 ISP는 입력 영상의 품질 개선을 위하여 영상 개선 알고리즘이 하드웨어로 설계되어 처리 속도는 빠르지만 다양한 영상 처리 알고리즘에 따라 성능 최적화에 어려움이 있었다. 제안한 ISP는 명령어를 기반으로 한 프로세서로서 다양한 영상 처리 알고리즘을 수행하고 SIMT 구조를 적용하여 알고리즘을 병렬로 수행해 성능을 개선하였다. 제안하는 ISP를 검증하기 위해 Xilinx Virtex-7을 탑재한 VC707 Board를 사용하였으며 cell multicore processor와 비교했을 경우 수행시간이 약 71%, ARM Cortex-A9과 ARM Cortex-A15와 비교하였을 경우 각각 63%, 33% 성능을 개선하였다.

Key words : SIMT, ISP, Pre-processing, Parallel processing, Processor

* LAON PEOPLE Inc.

** Dept. of Computer Engineering, Seokyeong University

★ Corresponding author

E-mail : kylee@skuniv.ac.kr Tel : +82-940-7745

※ Acknowledgment

This work was supported by the IT R & D program of Ministry of Trade, Industry and Energy(10049192, Development of a Smart Automotive ADAS SW-Soc for a Self-Driving Car)

Manuscript received Mar 2, 2016; revised Mar 21, 2016; accepted Mar 21, 2016

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. 서론

최근 컴퓨터 비전 기술들이 다양한 분야에 적용되고 영상 처리 기술에 대한 연구가 활발히 진행되고 있다. 스마트 모바일기기, 지능형 운전 보조 시스템 (Advanced Driver Assistance Systems)[1], 실시간 보안 시스템 등 영상 처리 기술이 상용화 되면서 영상 처리 기술에 대한 관심이 증가하고 있다.[2]

영상 처리 기술들은 다양한 변수들이 존재하는 입력 영상을 사용하기 때문에 정확한 수행을 위해 영상을 개선하는 ISP (Image Signal Processor)가 필요하다. ISP는 image sensor로부터 입력받은 영상 데이터에 대하여 전처리 과정을 수행한다. 입력 영상의 전처리 과정을 수행함으로써 사용자가 더 좋은 품질의 영상을 제공할 수 있으며 인식 기술에서도 향상된 성능을 보일 수 있다. 그리고 카메라 image sensor 기술이 발전하면서 최근 ISP는 고해상도 영상을 실시간으로 처리할 수 있는 연산 능력과 다양한 영상 전처리 과정들이 요구되고 있다. 하지만 기존의 ISP는 고정된 영상 처리 알고리즘을 수행하기 때문에 많은 영상 처리 알고리즘을 수행하기에는 한계가 존재한다. 이러한 문제를 해결하기 위해 본 논문에서는 다양한 영상 처리 알고리즘과 인식 알고리즘을 수행할 수 있는 명령어 기반의 ISP를 제안한다. 또한 하나의 프로세서로서 사용자가 요구하는 다양한 영상 처리 및 인식 알고리즘을 병렬로 처리하는 SIMT (Single Instruction Multiple Threads)^[3] 구조를 적용하여 ISP를 구현하였다.

이후 논문의 구성은 제안하는 ISP의 전체적인 구조와 지원하는 명령어 세트에 대해 소개하고 각 구조를 이루고 있는 sub-ISP에 대해서 설명하고자 한다.

II. 제안하는 SIMT 기반 Image Signal Processor

1. SIMT based ISP architecture

본 논문에서 제안하는 SIMT 기반 ISP는 입력 영상의 전처리 과정과 인식 알고리즘의 병렬 처리를 통해 고해상도의 영상에 대한 연산 가속화

를 지원한다.^[4] 제안하는 ISP의 구조는 그림 1과 같이 스레드들의 묶음인 warp의 개념을 가지며 총 16개의 스레드로 구성되어 있다. 즉 ISP는 총 16개의 Sub-ISP를 가지고 있다.

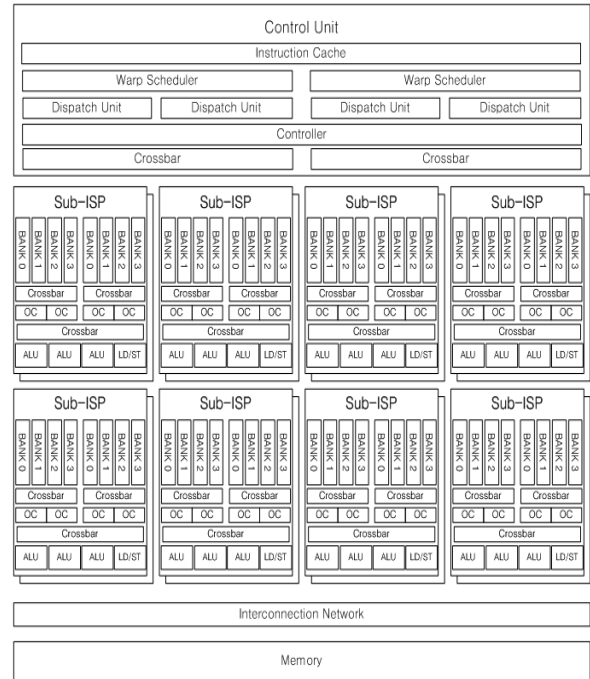


Fig. 1. The architecture of the proposed ISP

그림 1. 제안하는 ISP 구조

ISP는 홀수 warp과 짝수 warp의 명령어를 2개씩 받아 처리하며 슈퍼스칼라 구조를 사용하여 한 번에 최대 4개의 명령어를 instruction cache에서 가져올 수 있다. 발행된 명령어는 warp scheduler에 의한 중속성 검사를 통해 슈퍼스칼라 이슈가 가능한지 아닌지를 판단한다. 그리고 16개의 warp 중에서 스케줄링의 한 방식인 라운드 로빈 방식을 통해 선택된 홀수 warp과 짝수 warp의 명령어를 dispatch unit으로 전달한다. dispatch unit으로 전달된 명령어는 오퍼랜드의 인덱스나 오퍼랜드 활성 신호 등의 오퍼랜드 정보가 디코딩된다. 오퍼랜드에 대한 디코딩을 먼저 수행함으로써 레지스터를 참조하지 않는 오퍼랜드가 레지스터에 접근하게 되는 불필요한 상황을 방지하기 위한 방법이다.

2. ISP Instruction Set

제안하는 ISP는 명령어 기반 ISP로써 다양한 영상 처리 알고리즘과 인식 알고리즘 처리를 위하여 명령어 세트를 갖는다. ISP의 명령어는 크게 시스템 명령어 세트, 메모리 명령어 세트, 테

이터 이동 명령어 세트, 산술연산 명령어 세트로 총 4가지의 명령어 세트로 구성되어 있다. 각각의 명령어는 10가지의 인코딩 타입에 따라 분류된다.

시스템 명령어 세트는 ISP의 명령어 동작 흐름과 관련된 END, CALL JMP, RET과 같은 명령어로 구성되어 있으며 메모리 명령어 세트는 바이트, 워드, 하프워드 크기로 메모리와 Sub-ISP 레지스터 사이에서 데이터를 전송하는 LS, LU, SU, LDM, STM 명령어로 구성되어 있다. 데이터 이동 명령어 세트는 sub-ISP 내부의 레지스터의 데이터 이동 역할을 수행하며 상수 값에 대한 레지스터 이동도 가능하다. 바이트, 워드, 하프워드 크기로 데이터 이동이 가능하며 MV, MVU, MVI, MVUI와 같은 명령어로 구성된다. 마지막으로 산술연산 명령어 세트는 정수형 산술연산과 부동소수점 산술연산을 모두 지원 가능하며 레지스터와 레지스터 사이의 연산, 레지스터와 상수 사이의 연산을 모두 지원한다.

3. Sub-ISP

제안하는 ISP는 총 16개의 sub-ISP로 구성되어 있으며 16 warp, 16 thread로 동작한다. ISP 내부의 sub-ISP의 구조는 그림 2와 같다.

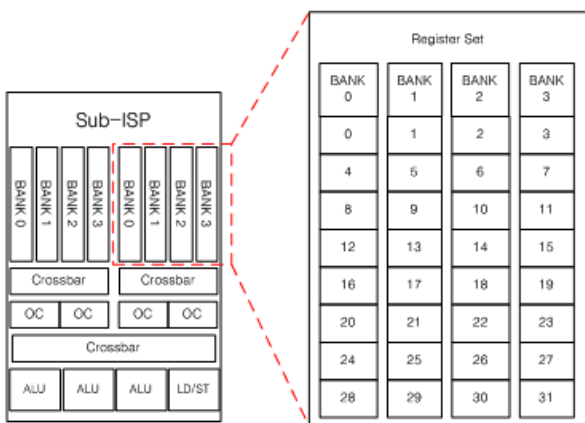


Fig. 2. The architecture of the sub-ISP
그림 2. Sub-ISP 구조

sub-ISP들은 각각 3개의 ALU(Arithmetic Logic Unit)와 1개의 LD/ST(Load/Store Unit), 그리고 8개의 Register Bank로 구성된 Register Set을 가지고 있다. sub-ISP는 홀수 warp과 짝수 warp에서 2개씩의 명령어를 받아 최대 3개의 ALU 연산과 1개의 메모리 연산을 동시에 처리할 수 있다. 또한 register bank로부터 명령어의

오퍼랜드가 참조하는 register에 대한 읽기/쓰기 동작을 수행하는 4개의 OC(operand collector)를 가지고 있다. OC 처리 이전에는 홀수 warp과 짝수 warp 2가지의 실행 흐름을 갖기 때문에 sub-ISP에서는 분리된 2개의 실행 흐름을 동시에 처리한다. 명령어를 동시에 처리할 때 발생할 수 있는 register conflict를 방지하기 위해 register가 bank마다 번갈아 가면서 배치되어있는 구조를 사용한다.

III 실험결과

본 논문에서의 ISP는 FPGA 환경을 통해 실험 및 검증은 진행하였다. FPGA 환경은 Xilinx Virtex-7을 탑재한 VC707 board^[5]를 사용하고 실험을 위해 Xilinx ISE 14.4와 Model-Sim SE 10.0b를 사용하였다. 또한 그림 3과 같이 Verilog HDL를 사용하여 제안하는 ISP를 구현하였으며 VC707 Board에 구현하기 위하여 AXI4와 AXI4-lite 시스템 버스 인터페이스를 사용하였다.

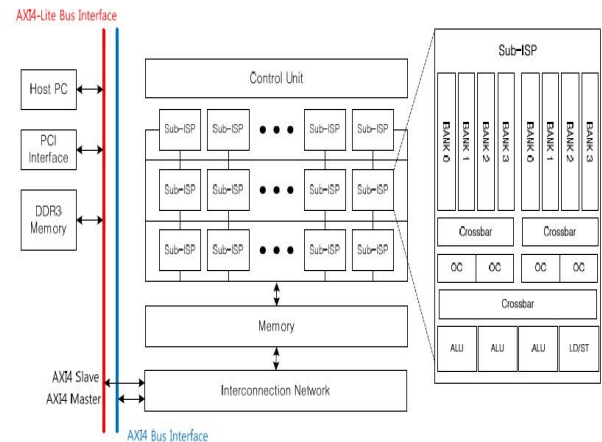


Fig. 3. Implementation of FPGA system using the proposed ISP

그림 3. 제안하는 ISP의 FPGA 시스템 구현

ISP 동작 확인을 위하여 외부 메인 PC의 CPU를 사용하였고 PCI-E 인터페이스를 이용하여 FPGA board의 DDR3 SDRAM에 데이터를 읽고 쓰는 방식을 사용하였다. ISP 최초 동작을 위해 메인 CPU에서 PCI-E 통신을 사용하여 초기 신호를 전달하면 ISP는 SDRAM에 저장된 명령어에 대한 처리를 시작한다. 명령어 처리 이후 데이터는 Board 메모리에 저장되고 PCI-E 통신으로 메인 PC의 메모리로 전송된다.

영상에 대한 FFT 전에 8 point에 대한 Radix-2 FFT를 수행하여 Matlab에서의 결과와 비교를 통해 정확도를 확인하였다. [82, 44, 62, 79, 92, 74, 18, 41]와 같은 8개의 임의의 값을 입력으로 FFT를 수행하였다. 그림 4은 Matlab과 ISP의 FFT 결과이다. FFT 수행 후 결과는 복소수 형태로 실수와 허수로 출력된다.

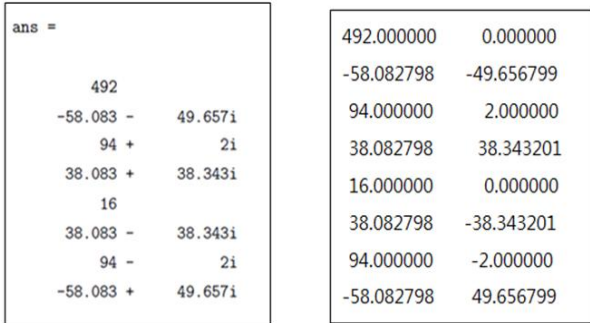


Fig. 4. The result of 8 point FFT(Left : Matlab, Right : ISP)

그림 4. 8 point FFT 결과 비교 (좌 : MATLAB, 우 : ISP)

제안하는 ISP는 16 warp, 16 thread로 동작하며 FPGA 플랫폼에서 동작 주파수는 50MHz이다. 성능 및 실험 결과를 위해 영상에 대한 FFT와 FFT를 사용한 edge enhancement으로 실험을 진행하였다.

Radix-2 FFT의 수행 시간 비교를 위해 1024 point에 대한 FFT 수행 시간을 표 1과 같이 cell multicore processor^[6], 임베디드 플랫폼의 CPU와 비교하였다. 1024 Point에 대한 FFT 수행 시간 비교 결과는 그림 5와 같다.

Cell multicore processor와 비교했을 경우 수행 시간이 약 71%, ARM Cortex-A9과 ARM Cortex-A15와 비교하였을 경우 각각 63%, 33% 감소한 것을 확인할 수 있다.

Table1. Processing time of Radix-2 FFT for 1024 points (ms)

표 1. 1024 point에 대한 Radix-2 FFT 수행 시간(단위 ms)

	Each of length	Time
Cell multicore processor	1024 points	2.64
ARM Cortex-A9	1024 points	2.02
ARM Cortex-A15	1024 points	1.12
Propose ISP	1024 points	0.74

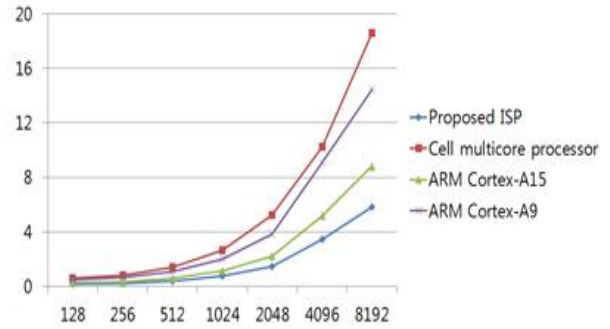


Fig. 5. The FFT processing time per point (axis x : the number of pixel point, axis y : processing time)

그림 5. Point 별 FFT 수행 시간 (x축 : 픽셀 포인트 수, y축 : 수행 시간)

FFT 연산 이후 2-D butterworth high pass filter를 사용하여 edge enhancement 과정을 수행하였다. 그림 6는 원본 영상과 butterworth high pass filter 그리고 edge enhancement된 영상이다.

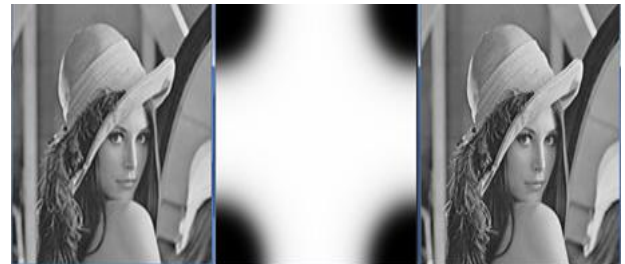


Fig. 6. The FFT result images of proposed ISP (Left : original image, Center: 2-D Butterworth high pass filter which is applied, Right: result image)

그림 6. 제안하는 ISP의 FFT 연산 결과 영상 (좌: 원본 이미지, 중: 적용된 2-D Butterworth High pass filter, 우: 결과 이미지)

IV 결론

본 논문에서는 기존의 ISP와 다르게 다양한 영상 전처리 알고리즘과 인식 알고리즘의 처리가 가능한 SIMT 기반 ISP를 제안하였다. 제안하는 ISP는 명령어 기반으로 영상 전처리 과정을 위해 사용되는 다양한 종류와 크기를 갖는 filter에 대한 적용이 가능하다. 제안하는 ISP는 효과적인 병렬 처리를 위해 16 warp, 16 thread로 동작하며 하나의 warp 당 16개의 sub-ISP로 구성되어 있다. 1024 point에 대한 FFT 수행 시간 비교 결과 cell multicore processor와 비교했을 경우 약 71%, ARM Cortex-A9과 ARM Cortex-A15와

비교하였을 경우 각각 63%, 33% 성능이 향상되었다.

References

- [1] NVIDIA, "Advanced driver assistance," <http://www.nvidia.com>
- [2] Ho-Sub Yoon, "The Trend of Technology and Market of Image Recognition Service," Communications of the Korea Information Science Society 31(2), pp. 23-31, 2013.
- [3] Erik Lindholm, John Nickolls, Stuart Oberman, John Montrym, "NVIDIA Tesla: A Unified Graphics and Computing Architecture," Micro IEEE vol. 28, pp. 39-55, 2008.
- [4] Jeon-yong Lee, "A Design of a Superscalar GP-GPU based on a Stream Processing for Smart Device," The Graduate School of Seokyeong University, 2014.2
- [5] Xilinx, "VC707 User Guide," <http://www.xilinx.com>
- [6] J. Barhen, T. Humble, P. Mitra, M. Traweck, "Multi-FFT Vectorization for the Cell Multicore Processor," IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, pp. 780-785, 2010.

BIOGRAPHY

Yun-seop Hwang (Member)



2014 : BS degree in Computer Engineering, Seokyeong University.

2016 : MS degree in Electronics Computer Engineering, Seokyeong University.

University.

2016 ~ present : Engineer, LAON PEOPLE Inc.

Hee-kyeong Jeon (Student Member)



2015 : BS degree in Computer Engineering, Seokyeong University.

2015 ~ present : MS candidate in Electronics Computer Engineering, Seokyeong University

University

Kwan-ho Lee (Student Member)



2015 : BS degree in Computer Engineering, Seokyeong University.

2015 ~ present : MS candidate in Electronics Computer Engineering, Seokyeong University

University

Kwang-yeob Lee (Life Member)



1985 : BS degree in Electronics Engineering, Sogang University

1987 : MS degree in Electronics Engineering, Yonsei University.

1994 : PhD degree in Electronics Engineering, Yonsei University.

1989 ~ 1995.2 : Senior Researcher, Hyundai Electronics Inc.

1995.3 ~ present : Professor, Dept. of Computer Engineering, Seokyeong University