

GPGPU를 이용한 H.264/AVC 디코더

김동한* · 이광엽*

*서경대학교

Implementation of IQ/IDCT in H.264/AVC Decoder Using GPGPU

Dong-han Kim* · Kwang-yeob Lee*

*Seokyeong University

E-mail : dhkim@skuniv.ac.kr

요 약

ITU-T와 ISO가 공동 제정한 동영상 압축 표준 H.264는 기존 동영상 압축 표준에 비해 높은 압축 성능과 유연성을 가진다. 본 논문에서는 병렬 처리에 효과적인 GPGPU(General-Purpose computing on Graphics Processing Units)를 이용하여 H.264/AVC 복호화 알고리즘에서 병렬 처리가 가능한 IQ/IDCT (Inverse Quantization/ Inverse Discrete Cosine Transform) 연산을 고속으로 수행하기 위한 효율적인 구조와 방법을 제안한다.

ABSTRACT

H.264/AVC(Advanced Video Coding) is a standard for video compression. H.264/AVC provides good video quality at substantially lower bit rates than previous standards. In this papers, we propose the efficient architecture of H.264/AVC decoder using GPGPU. GPGPU can process many of operation in parallel. IQ/IDCT is possible that parallel processing in H.264/AVC decoding algorithm.

키워드

GPGPU, GPU, H.264, Parallel Processing

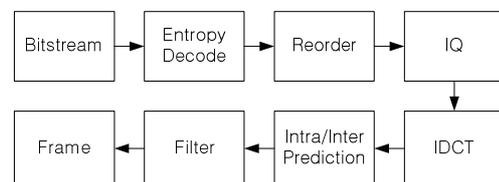
I. 서 론

최근 GPU(Graphics Processing Unit)의 연산 성능은 눈부시게 발전해왔다. 연산속도의 급격한 증가와 더불어 기존의 그래픽 애플리케이션의 연산뿐만 아니라 복잡한 과학 연산이나 암호 알고리즘 계산 등 그래픽 이외의 영역에서도 GPU의 사용을 가능하게 하였다. [1][2]

본 논문에서는 국제 표준인 H.264/AVC의 디코더 세부 기능에 대해서 설명하고 내부 파이프라인 중 병렬 처리가 요구되는 단계를 GP-GPU 명령어로 구현하였다.

검증을 위해서 Xilinx사의 Soft Core인 Micro Blaze를 CPU로 사용하였고 GP-GPU로 MTSP를 사용하였다. [3]

II. 본 론



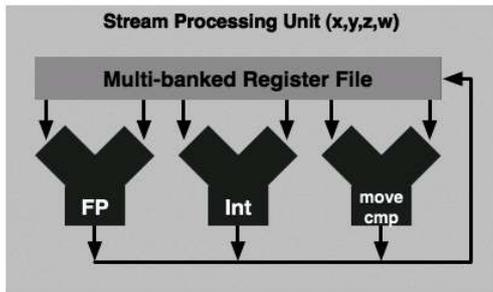
[그림 1]은 H.264 디코더의 블록 다이어그램이다. GP-GPU에서는 병렬 연산 방법이 매우 효율적인데 H.264 디코더에서 병렬 연산 방법을 이용할 수 있는 모듈은 Inter Prediction, IQ

(Inverse Quantization), IDCT (Inverse Discrete Cosine Transform) 이다 .

본 논문에서는 IQ와 IDCT 모듈을 GP-GPU로 구현하였고, 디스플레이에 필요한 CSC(Color Space Converter) 모듈도 GP-GPU로 구현하였다. GP-GPU를 이용하여 비디오 디코더를 구현하기 위해서는 GP-GPU의 로컬 메모리에 데이터 저장이 필요하며 본 논문에서는 GP-GPU가 직접 시스템 메모리에 접근하여 데이터를 로컬메모리에 저장하는 방법을 사용하였다. [4]

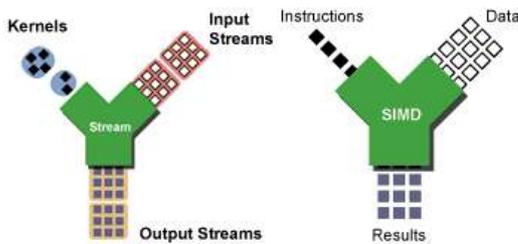
III. 병렬 처리 방법

GP-GPU로 데이터를 병렬 처리하는 방법은 GP-GPU안에 직접된 다수의 스트림 프로세서를 이용하는 방법이다. [그림 3]은 스트림 프로세서의 구조를 보여준다 .



[그림 2] 스트림 프로세싱 유닛

스트림 프로세서는 SIMD(Single Instruction Multiple Data)와 비슷하지만 좀더 발전된 형태이다. [그림 3]은 SIMD와 스트림 프로세서와의 차이를 나타내어준다 .

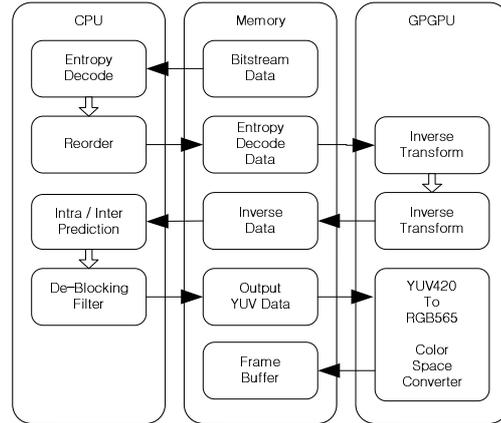


[그림 3] 스트림프로세서와 SIMD의 차이

본 논문에서는 H.264 디코더 알고리즘에서 매크로 블록단위로는 다수의 스트림 프로세서를 이용한 병렬 처리를 구현하였다 .

IV. 구현 및 성능

[그림 4]는 CPU와 GP-GPU에서 수행해야 할 작업들을 구분 하였다. CPU는 VLD코딩과 Intra/Inter Prediction, De-Blocking 필터, GP-GPU로 이동되는 데이터의 형식을 맞추는 역할을 수행한다. GP-GPU는 CPU로부터 전달된 데이터를 이용하여 IQ, IDCT, CSC 작업을 수행한다. 이 작업을 수행하기 위해서는 CPU와 GP-GPU 공유 메모리 설정이 중요하다 .



[그림 4] 디코더 구조

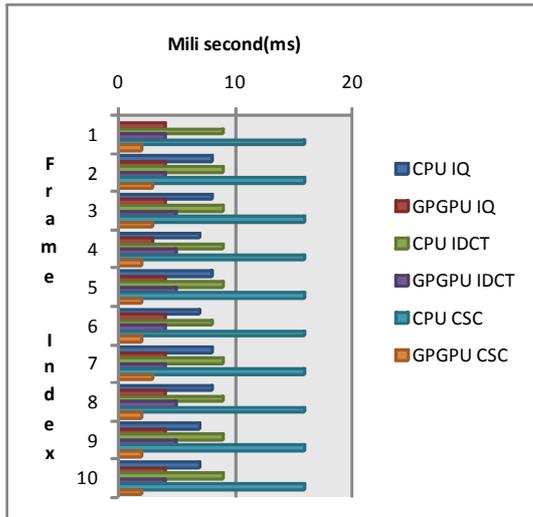
CPU에서 VLD 디코딩이 끝난 데이터들을 시스템 메모리에 저장하고, GP-GPU가 시스템 메모리에 접근하여 데이터를 로컬 메모리로 저장한다. 이후 IQ, IDCT를 처리하고 다시 시스템 메모리에 처리된 데이터를 저장한다. 시스템 메모리에 저장된 데이터를 이용하여 CPU는 Intra/Inter Prediction과 De-Blocking 필터를 처리하고 처리된 데이터를 시스템 메모리에 저장하면 다시 GP-GPU는 데이터 CSC후 시스템 메모리에 할당되어있는 프레임 버퍼에 저장한다.

[표 1]과 [그림 5]은 QCIF(176x144) 크기의 영상에 대한 각 모듈의 실험 결과를 나타낸 것이다. 모듈에 따라 CPU와 GP-GPU의 속도 차를 비교하였다. 비교결과 각 모듈에 대해서 CPU보다 GP-GPU로 연산을 수행할 경우 약 1.8배~7배정도의 성능 향상이 있다는 것을 알 수 있다 .

< 시간단위 : SEC >

모듈	CPU	GPGPU	Speed-up
IQ	0.0075	0.004	1.87
IDCT	0.009	0.0045	2
CSC	0.016	0.0023	6.96

[표 1] 각 모듈별 시간 비교



[그림 5] 각 모듈의 샘플 프레임 결과

성능 검증을 위한 환경으로 Xilinx사의 ML506 보드를 사용하였다. Virtex-5 FPGA에 Soft Core CPU인 Micro Blaze와 GP-GPU인 MTSP를 합성하여 125Mhz에서 성능 측정을 하였다.

V. 결론

본 논문은 H.264 디코더 내부 파이프라인 중 병렬처리가 가능한 부분을 CPU만으로 동작 시켰을 때보다 GP-GPU를 활용할 경우 보다 뛰어난 성능을 나타낸다는 것을 나타내었다. 이는 H.264 외에도 병렬 작업의 분량이 많은 영상처리 및 컴퓨터 비전 알고리즘을 처리할 경우 GP-GPU를 사용함으로써 성능 향상이 될 것으로 기대된다.

참고문헌

- [1] General Purpose GPU Programming (GPGPU) Website, <http://www.gpgpu.org>.
- [2] J. Owens et al., "A survey of generalpurpose computation on graphics hardware," Computer Graphics Forum, Vol.26, No.1, pp.80-113, March, 2007.
- [3] Hyung-Ki Jeong, "A Fully Programmable Shader Processor for Low Power Mobile Devices" Journal of IKEEE, Vol.13, No.2, February, 2009.
- [4] Ari hirvonen, Tapani lepanen, "H.263 Video Decoding on Programmable Graphics Hardware", IEEE international Symposium on Signal Processing and Information Technology, MAY, 2005