GPGPU 병렬 프로그래밍을 이용한 H.264/AVC 고속 화면내 예측 모드 결정

*최성준 **한기훈 ***유영수

(주) 필링크 연구소

*seewoo@feelingk.com **khhan@feelingk.com ***ysyoo@feelingk.com

H.264/AVC Fast Intra Mode Decision using GPGPU Parallel Programming

*Sung-Jun Choi **Ki-Hun Han ***Yeong-Soo Yoo

Feelingk R&D Center

요약

GPU의 병렬성과 연산능력을 일반적인 공학적 문제 해결에 적용하는 GPGPU 컴퓨팅에 대한 연구가 최근 활발히 진행되고 있다. 비디오 압축과정에는 많은 양의 화소 데이터에 동일하게 반복되는 연산을 수행하는 알고리즘이 많이 적용되므로 GPGPU를 통한 고속 병렬 계산의 응용 분야로 매우 적합하다. H.264/AVC는 비디오를 압축하는 가장 최신의 국제표준으로 여러 제품군과 서비스에 대한 적용되어 시장에서 널리 사용되고 있다. 본 논문에서는 GPGPU의 응용 분야로 주목 받고 있는 비디오 압축 분야에 대한 적용으로 H.264/AVC의 화면내 예측 모드 결정과정에 GPGPU 병렬 프로그래밍을 적용하여 예측 모드 결정 속도를 향상하는 방법을 제안한다. GPU상에서의 데이터 병렬처리를 위해 CUDA C언어를 사용하였으며, CPU상에서의 연산은 C언어를 사용하여 구현되었다. GPU상에서 프레임 전체에 대한 화면내 예측 모드를 병렬적으로 결정함으로써 이에 소요되는 시간을 줄여 줄 수 있었다. 실험결과 GPU상에서 병렬적으로 예측 모드를 결정할 때 Full-HD급 영상에서 약 2.8배 정도의 속도 향상을 확인할 수 있었다. 향후 GPGPU 병렬 프로그래밍을 화면 내 예측뿐만 아니라 반복되는 연산을 수행하는 다른 알고리즘에도 적용하여 부호화기의 계산 부담을 덜어준다면 고속 실시간 비디오 압축 부호기 개발이 더욱 용이해 질것으로 기대된다.

1. 서론

병렬 응용 프로그램의 처리량을 늘려주기 위해 작은 코어를 대량으로 사용하는 매니코어(many-core) 방식에 따른 GPU 개발 방식이 등장함에 따라 GPU 하드웨어 설계가 더욱 더 고성능 병렬 컴퓨팅 기능을 지원하는 방향으로 빠르게 발전하였다[1]. 이에 많은 연구자들이 GPU를 기존의 그래픽처리에 한정되어 사용하는 것에서 벗어나, GPU를 이용하여 많은 계산량을 필요로 하는 일반적인 과학 및 공학적 문제를 해결하는 가능성을 연구하였다. 이러한 방법을 GPGPU(General Purpose GPU)라고 부르며, 간단하게 말해서 GPU가 기존 CPU의 연산을 대신 수행하는 것이다. GPGPU[2]의 응용 분야로는 객체인식, 분자역학, 패턴분석, 양자화학, 회로설계, 영상처리 등 대용량 데이터를 병렬적으로 처리하는 여러 분야를 들 수 있다. 비디오 압축은, 구성하는 다수의 알고리즘이 많은 양의 화소 데이터에 동일하게 적용되는 연산을 포함하고 있어, GPGPU의 응용 분야로 주목 받고 있다.

비디오 압축 국제표준은 ITU-T VCEG과 ISO/IEC MPEG에 의해 발전해 왔으며, VCEG에서는 H.261, H.263 등의 표준을 제정하였고, MPEG에서는 MPEG-1, MPEG-2, MPEG-4 등의 표준을 제정하였다. 가장 최신의 비디오 압축 국제표준은 VCEG과 MPEG의 공동작업으로 만들어진 H.264/AVC[3] 이며, 기존의 압축 표준에 비해 월등한 성능을 보여 시장에서 널리 사용되고 있다.

본 논문에서는 H.264/AVC 의 화면내 예측모드 결정을 GPU에서 병렬적으로 수행하여 속도를 향상하는 방법을 제안한다. GPU상에서의 데이터 병렬 처리는 NVIDIA사에서 발표한 CUDA C[4-5]언어로 구현하였다. GPU상에서의 데이터 병렬 처리 방식으로 화면 내 예측 모드를 결정할 때 프레임 전체에 대해서 한꺼번에 처리함으로써, CPU에서 순차적으로 예측 모드를 결정할 때에 비해 Full-HD급 영상에서 약 2.8배 정도의 속도 향상을 확인 할 수 있었다. 본 논문에서는 비록 화면 내 예측 모드 결정에서만 GPGPU를 적용한 속도 향상을 확인 하였지만, 비디오 압축에는 화면 내 예측 외에도 움직임 추정/보상, 부화소 보간, 변환 및 양자화 등과 같이 다수의 데이터에 동일한 연산을 수행하는 다수의 알고리즘이 존재한다. 향후 이러한 알고리즘을 GPU상에서 구현한다면 부호화기의 계산 부담을 덜어주어 고속 실시간 비디오 압축 부호기 개발이 더욱 용이해 질 것으로 기대된다.

2. GPU구조 및 시스템 구성

GPU는 CPU에 비해 훨씬 많은 트랜지스터를 연산을 위해 할당하고 병렬처리에 특화된 구조를 가지고 있다. CUDA를 이용한 GPU 병렬 처리 프로그래밍의 실행단위는 그림 1과 같이 다수의 Block과 Thread로 이루어진다. 하나의 Device는 다수의 블록으로 구성되고, 각각의 Block은 다수의 Thread로 구성된다. 그림에서는 Device가 간단히 몇 개의 Block과 Thread로 구성되었으나 실제 응용 시 Thread의 개수는 수천 ~ 수백만 개에 이른다. Thread의 개수가 많을수록 병렬성이 증가하지만, GPU의 메모리 구조와 처리되는 데이터의 의존, 알고리즘 구성 등을 모두 고려하여 Block과 Thread 개수를 적절히 정해주어야 한다.

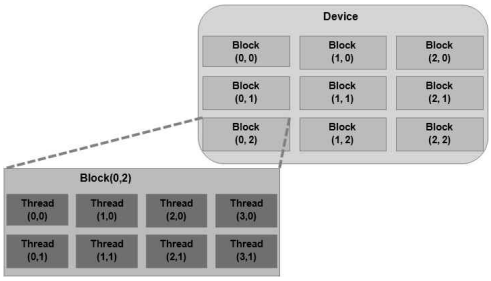


그림 1. GPU 병렬 처리 프로그래밍의 실행단위: Block, Thread.

GPU의 계산력을 활용하는 GPU 가속 코덱 시스템의 구성도는 그림 2와 같다. 기존에 CPU(Host)에서 이루어지던 연산 중 일부를 GPU에서 실행한다. 이를 위해 CPU에서 GPU로 비디오 프레임의 YUV 값과 같은 데이터와 필요한 Coding Parameter 등을 전송하고 GPU는 많은 코어를 기반으로 동시에 많은 Thread를 병렬적으로 수행하여 빠른 연산을 수행하고, 연산의 결과를 다시 CPU로 전송해 주는 방식이다.

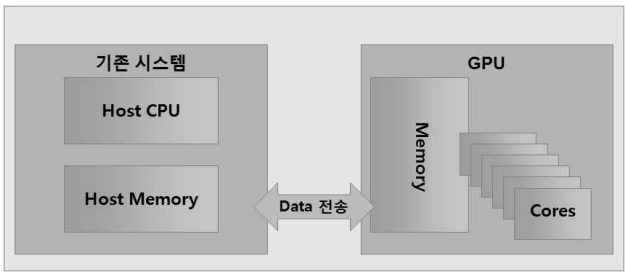
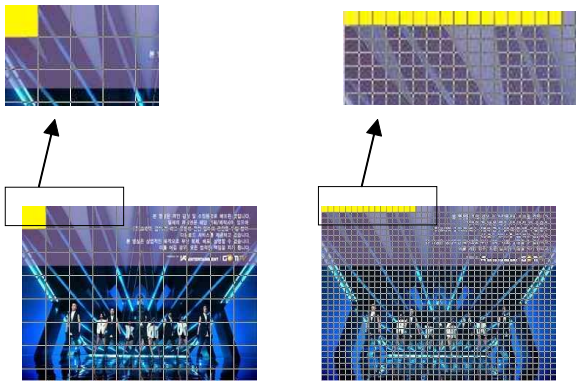


그림 2. GPU 가속 코덱 시스템 구성도

본 논문에서는 H.264/AVC의 부호화 과정에서 화면내 예측 모드 결정 과정을 GPU에서 수행하였다. CPU에서 전체 프레임의 YUV 데이터를 GPU로 전송해주고, GPU에서 해당 프레임 전체의 화면내 예측 모드를 결정하여 모드 정보를 다시 CPU로 전송해주면, CPU에서는 이후의 모든 부호화 과정을 수행하여 압축 비트스트림을 생성하는 방식으로 코덱이 구동된다.

3. 고속 병렬처리를 위한 화면 내 예측방법

H.264/AVC Baseline Profile에는 휘도 성분에 대해 Intra16x16, Intra4x4 모드와 같이 2가지 예측 블록 크기를 지원한다. Intra16x16 모드에서는 매크로 블록 전체가 하나의 16x16 블록 크기로 예측되고, Intra4x4 모드에서는 매크로 블록이 16개의 4x4 블록으로 나누어져, 각4x4 블록 크기로 예측된다. Intra16x16 예측 모드는 Vertical, Horizontal, DC, Plan 등과 같이 4가지가 있고, Intra4x4 예측 모드는 수직, 수평, 대각 방향 등 8가지 방향별 예측 모드와 DC 모드를 포함하여 9가지가 있다.



(a) Intra 16x16 Thread Block (b) Intra 4x4 Thread Block

그림 3. 화면 내 예측 Thread, Block 구성

본 논문에서는 화면 내 예측 프레임(I 프레임)에서 Intra16x16 모드와 Intra4x4 모드 부호화 시, 여러 예측 모드 중 최적의 모드를 결정하는 과정을 GPU상에서 병렬적으로 처리 하였다. 전체 프레임에서 Intra16x16 모드의 최적 예측 모드를 결정하기 위해 화면을 구성하는 휘도 성분의 화소수 만큼의 Thread를 생성하였으며 모든 Thread들은 매크로 블록 단위로 Block으로 묶어 병렬적으로 처리하였다. 다시 말해, GPU에서 실행되는 Block의 개수는 프레임의 매크로 블록 개수와 같으며 각각의 Block은 256(16x16)개의 Thread로 구성되었다. 각 Thread에서 예측 모드 별 error값을 구하고, Block내에 속한 모든 Thread들의 error값을 더하여 매크로 블록의 예측모드별 error값을 구한 후, Intra16x16 모드의 예측모드를 결정하였다.

Intra 4x4 모드의 예측 모드 결정은 Intra16x16과 Block 구성에서 차이가 있다. 화면을 구성하는 휘도 성분의 화소 수만큼의 Thread를 구성하는 것은 Intra16x16과 동일하지만 Thread들을 Block으로 구성할 때 차이가 있다. Intra4x4 모드의 경우, 4x4 블록 단위로 예측 모드가 결정되므로 4x4화소에 해당하는 Thread 16개를 하나의 Block으로 구성할 수 있지만, 이 경우 개별 Block이 실행될 때 16개의 Thread만 동작하므로 GPU 내부의 Thread 집적도가 낮아져 GPU 내부의 다수의 스트리밍 멀티프로세서 중 일부분만 사용하게 되는데 이는 GPU 병렬연산의 효율성을 떨어뜨린다. 이를 해결하기 위해 Block 실행 시 보다 많은 병렬성을 확보해야 되는데, 다시 말해 실행되는 Thread의 개수를 증가 시켜야 하는데 본 논문에서는 그림 3의 (b)와 같이 동일 Line의 4x4 블록 16개를 1개의 Thread Block으로 설정하였다. 4x4 블록 16개를, 다시 말해 256개의 Thread를 1개의 Block으로 묶음으로써 4x4 블록 단위로 16개의 Thread를 하나의 Block으로 설정할 때보다 약 2배의 성능향상을 얻을 수 있었다. Intra4x4 모드에서도 Intra16x16 모드에서와 동일하게 각 Thread에서 예측방향별 error값을 구한 후 Thread들의 error값을 4x4 블록 단위로 모아 각 블록의 예측모드를 결정하였다.

4. 실험결과

본 논문에서는 실제 비디오 스트리밍 서비스에 사용되는 동영상(그림 3)을 VGA (640 x 480), HD (1280 x 720), FHD (1920 x 1080)로 변환하여 테스트를 진행하였다. 실험 환경은 표 1 과 같으며, GPU 상에서의 화면내 예측을 구현하기 위한 개발언어로는 NVIDIA사의 CUDA C를 사용하였다. 테스트를 위한 코드는 C언어로 개발한 참조 코드와 GPU 연산 기능이 추가된 코드를 사용하였다.

표 1. 실험 환경

항목	세부사항
CPU	Intel® i7-920 2.67GHz
GPU	Nvidia Geforce GTX 560Ti
RAM	8.0 GB
OS	Microsoft Windows 7
Tool	Microsoft Visual Studio
	CUDA 4.0

CPU 만을 사용한 결과와 GPGPU 병렬 프로그래밍을 적용하였을 때의 성능을 비교한 결과는 표 2 같다. 본 논문에서 제안한 GPGPU를 적용한 고속 화면 내 예측 방법은 속도면에서 CPU 만을 사용한 화면 내 예측 결과보다 실행속도가 1.75 ~ 2.8배 빨라지는 것을 확인 할 수 있다. 그리고 영상의 해상도가 높을 수록 매크로 블록의 개수가 많아지고 이로 인해 동시 처리할 수 있는 Thread의 개수가 높아지므로 속도 향상이 커지는 것을 알 수 있다.

표 2. 연산속도 비교

항목	화면 내 예측 속도 (fps)		
	VGA	HD	FHD
CPU Only	360.35	120.43	53.31
GPGPU ON	629.49	295.52	149.03
속도비	1:1.75	1:2.5	1:2.8

비디오 압축은 점차 고해상도의 컨텐츠를 압축하는 방식으로 발전하고 있어, 향후 더욱 많은 화소 수를 가진 프레임을 병렬적으로 처리 할 때 GPGPU를 적용하면 더욱 좋은 성능향상을 얻을 것으로 예상된다.

5. 결론

GPU를 범용으로 이용하는 GPGPU 컴퓨팅에 대한 연구가 최근 활발히 진행되고 있다. 본 논문에서는 GPGPU의 한 응용분야로 주목받고 있는 비디오 압축에서 화면 내 예측 모드 결정에 대한 부분을 GPU에서 수행함으로써 기존의 CPU만 이용하였을 때와 성능을 비교하여 비디오 압축에서의 GPGPU 응용 가능성을 확인하였다. 본 논문에서 구현한 Intra 예측 모드 결정 외에도 비디오 압축에는 움직임 추정/보상, 부화소 보간, 변환 및 양자화 등 많은 데이터를 병렬적으로 처리 할 수 있는 알고리즘이 다수 존재 한다. 또한 비디오 압축의 최근 발전 방향을 보면 점차 많은 프레임 화소 수를 가진 컨텐츠를 압축하는 방향으로 진행되고 있어 향후 GPGPU를 응용할 수 있는 분야로써 비디오 압축이 매우 발전 가능성이 많은 분야라 할 수 있다. 본 논문에서 제안한 화면 내 예측 모드 결정 알고리즘에서의 가능성을 향후 다른 알고리즘으로 확장하여 고속 실시간 부호화기 개발에서의 GPGPU 병렬 컴퓨팅 기술이 한 축을 담당할 것을 기대 할 수 있다.

감사의 글

“본 논문은 중소기업청에서 지원하는 2011년도 산학연공동기술개발사업(No. 00044957)의 연구수행으로 인한 결과물임을 밝힙니다.”

참고문헌

[1] “대규모 병렬 프로세서 프로그래밍”, 하순회, 김크리스, 이영민 옮김, BJ퍼블릭, 20011년 2월.
[2] http://www.gpgpu.org/
[3] ITU-T Reccommendation H.264 and ISO/IEC 14496-10, "Advanced video coding for generic audiovisual services," May 2003.
[4] "CUDA BY EXAMPLE", J. SANDERS, E. KANDROT, Addison Wesley
[5] http://developer.nvidia.com/category/zone/cuda-zone