

## 로드밸런싱 기반의 HEVC 디블록킹 필터 병렬화 방법

\*류호찬, 정순홍

\*디지털인사이트 주식회사, 한국전자통신연구원

\*hc.ryu@digitalinsights.co.kr, zeroone@etri.re.kr

### Load Balancing based Parallelization Method of HEVC Deblocking Filter

\*Hochan Ryu, Soon-heung Jung

\*Digital Insights Inc., Electronics and Telecommunications Research Institute

#### 요 약

본 논문에서는 로드밸런싱 (load balancing) 기반의 HEVC (High Efficiency Video Coding) 디블록킹 필터 (deblocking filter) 병렬화 방법을 제안한다. 본 논문에서는 디블록킹 필터의 병렬화를 위해 TU (Transform Unit)의 경계 정보를 이용하여 픽처 단위로 디블록킹 필터의 작업량을 예측하고, 예측된 작업량을 기반으로 코어에 균등한 일을 할당함으로써 디블록킹 필터의 병렬화 효율을 향상 시켰다. 실험결과 제안하는 로드 밸런싱 기반 디블록킹 필터 병렬화 방법을 사용하여 균등 분할 디블록킹 필터 병렬화 방법에 비해 11%의 디블록킹 필터 속도를 향상시켰다.

#### 1. 서론

하드웨어 장치 및 소프트웨어가 발달함에 따라 고해상도, 고화질의 비디오에 대한 시장의 수요가 늘고 있다. 이에 고해상도 비디오를 효과적으로 압축할 수 있는 압축 표준이 필요하게 되었고, ISO/IEC의 MPEG과 ITU-T의 VCEG은 공동으로 JCT-VC를 결성하여 HEVC (High Efficiency Video Coding)을 개발하였다 [1]. HEVC의 디블록킹 필터 (deblocking filter)는 블록 단위 양자화 수행에 따른 블록화 현상을 제거하기 위해 인-루프 필터 형태로 복호화된 영상에 대해 수행됨으로써 주관적 화질을 향상시킨다. 고해상도 및 고화질 비디오에 대한 실시간 부호화 또는 복호화를 위해 다수의 코어를 이용하여 병렬처리하여 부호화 또는 복호화하는 방법들이 연구되고 있으며, 디블록킹 필터 또한 다수의 코어를 사용하여 병렬처리 가능하다. 디블록킹 필터를 병렬화하기 위해서 일반적으로 영상을 균등한 영역으로 분할 한 후, 각 영역을 코어 또는 스레드에 할당하여 필터링을 수행하는 균등분할 병렬화 방법을 사용한다. 하지만 균등분할 병렬화 방법은 균등하게 분할된 각 영역의 일의 양이 서로 달라 병렬화 효과가 극대화되지 않을 수 있다. 이러한 문제를 해결하기 위해 디블록킹 필터의 작업량을 예측하여 동일한 작업을 각 코어에 할당함으로써 병렬화 효율을 향상시킬 수 있다. 본 논문에서는 TU (Transform Unit)의 경계의 수를 측정하여 디블록킹 필터의 작업량을 예측한 후, 동일한 작업량을 가지도록 영상을 분할하여, 분할된 각 영역을 코어에 할당하는 로드밸런싱 기반의 디블록킹 필터 병렬화 방법을 통해 병렬화 효율을 향상시켰다.

본 논문의 구성은 다음과 같다. 2절에서는 HEVC의 디블록킹 필터 기술과 병렬화 방법에 대해 살펴본 후, 3절에서는 제안하는 디블록킹 필터 병렬화 방법을 소개한다.

4절에서는 제안하는 방법의 성능을 실험을 통해 살펴보고, 마지막으로 5절에서 본 논문에 대한 결론을 맺는다.

#### 2. HEVC 디블록킹 필터 기술과 병렬화 방법

비디오 부호화 과정에서 블록 단위 양자화 수행에 따른 여러로 인해 블록화 현상이 발생한다. 이러한 블록화 현상은 주관적 화질에 큰 영향을 미치기 때문에 이를 제거하기 위하여 HEVC는 인-루프 필터 형태의 디블록킹 필터를 사용한다. HEVC의 디블록킹 필터는 CTU 단위로 수행되며, 화면 단위로 수직 경계에 대해서 수평 필터링이 먼저 수행된 후, 수평 경계에 대한 수직 필터링이 적용된다. 화면 단위로 수직 경계에 대해 먼저 필터링하고 수평 경계에 대해서 필터링하는 것은 H.264/AVC에서 매크로블록 단위의 수평 및 수직 필터링이 번갈아가 가면서 적용되던 것에 비해 인접 블록간의 의존성을 제거할 수 있어 데이터 레벨 병렬화를 적용하기에 효과적이다 [2]. HEVC는 디블록킹 필터링을 취하기 위해서 가장 먼저 필터를 적용해야하는 경계 후보를 결정한다. 하나의 CU 내에서  $8 \times 8$  격자 위치에 존재하는 PU 또는 TU 경계가 디블록킹 필터의 수행이 필요한 경계이다.

데이터 레벨 병렬화 방법은 전체 데이터 량을 분할하여 각 코어에 할당한다. 데이터 레벨 병렬화 방법에서 병렬화 효율은 데이터 분할 및 코어에 할당하는 방법에 따라 결정된다. 데이터 레벨 병렬화에서 데이터의 양에 따른 일의 양이 비례할 경우, 일반적으로 균등 크기의 데이터 분할 방식을 사용한다. 하지만 모든 데이터가 동일한 일을 수행하지 않을 경우에 데이터를 균등하게 분할할 경우에는 코어 간의 작업량 불균형으로 인해 병렬화 효율이 저하된다. 그러므로 코어에 일의 양을 균등하게 분배하면 동일시기에 모든 코어의 일이 끝나게 되므로 병렬화 효율을 높일 수 있다. 따라서 데이터의 일의 양을 예측하여 각

코어에 균등하게 일의 양을 할당하는 과정이 필요하다.

### 3. 제안하는 디블록킹 필터 병렬화 방법

HEVC 의 디블록킹 필터는 양자화 계수의 존재 유무, 참조 프레임 인덱스, 움직임 벡터의 차이, 블록 분할 정보에 따라 필터링 여부가 결정된다. 이러한 필터링 여부를 결정하는 과정은 실제 디블록킹 필터링 과정에서 많은 부분을 차지하므로, 이를 바탕으로 코어에 할당할 분할 영역을 결정하기에는 필터링 여부를 결정하는 과정의 연산량이 높아 병렬화 효율이 떨어진다. 병렬화 효율을 높이면서 작업량을 평가하기 위해서는 작업량 예측을 위해서 보다 간단한 상관성 있는 파라미터를 찾고 이를 통해 작업량을 예측해야 한다.

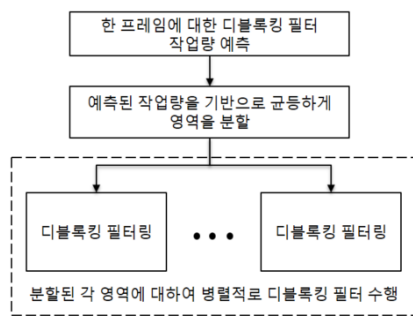


그림 1. 제안하는 병렬화 방법의 수행 순서

제안하는 방법에서 디블록킹 필터의 작업량을 예측하기 위해  $8 \times 8$  TU 경계의 수를 측정한다. HEVC 의 디블록킹 필터에서는  $8 \times 8$  격자 위치에 존재하는 PU 또는 TU 경계에 대하여 필터를 수행하므로,  $8 \times 8$  TU 경계의 수는 디블록킹 필터 경계 후보와 상관성이 높을 것이다. 제안하는 방법의 과정은 그림 1 과 같다. 먼저 한 프레임에 대하여 디블록킹 필터링의 복잡도를 계산한다. 그리고 계산한 복잡도를 기반으로 코어에 할당할 CTU 의 구역을 계산한 후, 각 코어에 CTU 를 할당하여 디블록킹 필터링을 병렬적으로 수행한다. 제안하는 방법에서는 그림 2 와 같이  $8 \times 8$  TU 경계의 수를 이용하여 디블록킹 필터의 작업량을 예측하여 영상을 균등하게 분할하였다.  $32 \times 32$  TU 내에는 8 개의  $8 \times 8$  경계가 존재하고,  $16 \times 16$  TU 내에는 4 개의  $8 \times 8$  경계가 존재,  $8 \times 8$  TU 내에는 2 개의  $8 \times 8$  경계가 존재한다.

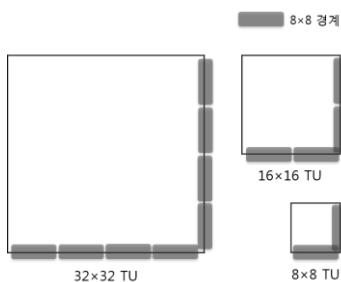


그림 2. TU 내의  $8 \times 8$  경계의 수

### 4. 실험 결과

제안하는 디블록킹 필터 병렬화 방법의 성능을 측정하기 위해 HM12.0 참조 소프트웨어를 기반으로 OpenMP[3]를 사용하여 제안하는 병렬화 방법을 구현하였다. 병렬화 방법의 성능향상 정도를 측정하기 4 개의 코어를 사용하여 한 개의 코어를 사용했을 때와 비교하여 몇 배 속도향상이 있는지를 각 병렬화 방법마다 측정하였다. 제안하는 디블록킹 필터 병렬화 방법의 성능은 표 1 과 같다. P1 은 균등 분할 디블록킹 필터 병렬화 방법이며, P2 는 제안하는 로드밸런싱 기반 디블록킹 필터 병렬화 방법이다. P2 방법이 P1 방법에 비해 평균 11% 병렬화 성능 향상이 있는 것을 확인할 수 있다.

표 1. 실험 결과

Sequence	speed-up	
	P1	P2
BasketballDrive	2.76	3.31
BQTerrace	3.01	3.24
Cactus	2.65	3.06
Kimono	3.14	3.56
ParkScene	3.07	3.05
Average	2.92	3.25

### 4. 결론

본 논문에서는 TU 경계의 수를 측정하여 작업량을 예측한 후, 코어에 균등한 작업을 분배하여 디블록킹 필터 병렬화 효율을 향상시키는 방법을 제안하였다. 제안하는 방법을 통해 균등 분할 디블록킹 필터 병렬화 방법에 비해 11%의 병렬화 성능 향상을 얻었다.

### 감사의 글

이 논문은 2015 년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (R0101-15-0283, 클라우드 기반 UHD 방송콘텐츠 스트리밍 서비스 기술 개발)

### 참고문헌

[1] High Efficiency Video Coding, Rec. ITU-T H.265 and ISO/IEC 23008-2, Jan 2013.

[2] A. Norkin, G. Bjøntegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. Auwera, "HEVC deblocking filter," IEEE trans. on Circuits and Systems for Video Technol., vol. 22, no. 12, pp. 1746-1754, Dec. 2012.

[3] L. Dagum and R. Menon, "OpenMP: an industry standard API for shared-memory programming," IEEE Computational Science & Engineering, vol. 5, no. 1, pp.46-55, Jan. 1998.