

논문 2016-11-27

모바일 환경에서 HEVC 인트라 인코딩의 계산 복잡도 감소를 위한 영상 특성 기반의 블록 후보 조기 결정 방법

(Texture-based Early Decision of Block Sizes for the
Complexity Reduction of HEVC Intra-Encoding
in the Mobile Environment)

박승원, 이채은*

(Seung-Won Park, Chae Eun Rhee)

Abstract : Compared to the former H.264 standard, the number of the prediction modes has highly increased in HEVC intra prediction. Compression efficiency and accurate prediction are significantly improved. However, the computational complexity increases as well. To solve this problem, this paper proposes the new scheme where not only prediction modes but also block partition candidate are early chosen. Compared to the original intra prediction in HEVC, the proposed scheme achieves about 38% reduction in processing cycles with a marginal loss in compression efficiency.

Keywords : Video coding, HEVC, Intra prediction, Encoder, Early mode decision

I. 서론

반도체, 통신, 영상 처리 기술 등이 발전하면서 고화질 비디오에 대한 요구가 급격하게 증가하였다. 하지만 고화질의 비디오 표현을 위해서는 필요한 데이터 용량이 매우 크다. 영상을 획득, 처리, 저장하거나 전달하기 위해 높은 계산 복잡도, 큰 저장 공간 및 전력 소모 등이 요구된다. 따라서 모바일 환경과 같이 전력 공급에 제한이 있고, 프로세서와 하드웨어 가속기, 메모리 등의 성능 제약이 있는 환경에서는 고화질 비디오를 이용하는데 한계가 있다. 이러한 한계를 극복하기 위해 고해상도 비디오를 효과적으로 압축하여 획기적으로 데이터의 양을

줄이는 많은 기술의 발전들이 있어왔다. 2013년 1월에는 기존의 H.264/AVC 비디오 코덱 표준 보다 월등한 압축 성능을 갖는 High Efficiency Video Coding(HEVC)이 차세대 비디오 코덱 표준으로 선택되었다. HEVC는 H.264/AVC와 비교하여 동일한 수준의 화질을 얻기 위해 절반의 대역폭만을 사용하는 것을 목표로 하고 있다. 하지만 HEVC는 H.264/AVC와 비교하여 인코딩 블록 단위가 64x64~4x4까지 다양해지고 인트라 예측 모드 수는 약 4배 만큼 증가함에 따라 계산 복잡도가 크게 높아졌다.

H.264와 비교하여 인트라 예측 모드가 다양해짐에 따라 모드 결정이 최종 압축률에 크게 영향을 미친다. H.264 인코더에서 많이 사용되어 왔던 distortion 기반의 간략화된 모드 결정 기법을 사용할 경우 압축 효율이 크게 떨어진다. 따라서 최적의 예측 모드와 블록 크기를 결정하기 위한 Rate-Distortion Optimization(RDO) 단계가 인트라 예측의 계산 복잡도에서 큰 비중을 차지하게 되었다. HEVC 참조 소프트웨어인 HEVC test Model(HM)에서는 RDO의 계산 복잡도를 낮추기 위해 예측 모드들의 하다마드 변환 기반의 Sum of

*Corresponding Author (chae.rhee@inha.ac.kr)

Received: 21 June 2016, Revised: 12 July 2016,

Accepted: 21 July 2016.

S.W. Park, C.E Rhee: Inha University

※ 이 논문은 2015년도 정부(미래창조과학부)의
재원으로 한국연구재단의 지원을 받아 수행된 기
초연구사업임(NRF-2015R1C1A1A02037625).

Absolute Transformed Differences(SATD) 값을 비교하여 RDO를 수행하는 모드의 수를 줄이는 방법인 Rough Mode Decision(RMD) 단계를 도입하였다. 하지만 RMD를 통해 줄어드는 계산 복잡도의 양으로는 충분하지 않기 때문에 인트라 예측의 계산 복잡도를 줄이기 위한 다양한 연구들이 있어왔다.

논문 [1]에서는 RDO에서의 discrete cosine transform(DCT) 과정을 하다마드 변환으로 대체하여 RDO 단계 내에서의 계산 복잡도를 감소시켰다. 논문 [2]에서는 계산 복잡도 감소를 위해 RDO를 수행하는 예측 모드의 수를 줄이는 방법을 적용하였다. 하지만 이러한 두 가지 방법을 이용한 접근 방법은 압축 효율이 크게 저하된다는 한계가 있다. 논문 [3]에서는 RDO에서의 cost 계산 단순화를 통해 계산 복잡도를 감소하였다. 하지만 이 방법은 전처리 단계를 추가함으로써 인코딩 처리의 지연 시간이 늘어난다. 또한, 계산 복잡도 감소량에 비해 압축 성능의 저하가 매우 크다는 단점이 있다. HM에 적용되어 있는 RMD 단계처럼, RDO를 수행하기 전 예측 모드의 수를 줄여 계산 복잡도를 개선하고자 하는 많은 선행 연구들이 존재하였다. 하지만 HEVC에서 제공하는 다양한 크기의 인코딩 블록 중에서 영상 특성을 기반으로 선택된 가장 적합한 인코딩 블록 크기를 RDO 수행 전 미리 찾을 수 있다면 RDO 단계에서의 계산 복잡도는 크게 감소할 것이다.

본 논문에서 제안하는 Partition Filtering (PF) 알고리즘은 영상 특성을 기반으로 적합한 블록 사이즈의 후보군을 미리 선정하여 RDO를 수행한다. RDO 단계의 입력이 되는 블록의 수를 줄임으로써 계산 복잡도를 줄이는 방법이다. 제안하는 PF 기법은 filtering 단계에서 블록의 처리 순서를 따르지 않고 병렬적으로 처리를 한다. 때문에 추가되는 인코딩 시간 오버헤드는 매우 적다. 단, 병렬적인 처리가 가능한 하드웨어 기반의 인트라 인코더에 적합하다. 적용 가능성과 최종 성능을 논문 [4]에서 제안하는 인트라 예측 하드웨어 아키텍처를 사용하여 보인다.

본 논문의 2장에서는 제안하는 PF 알고리즘에 대해 자세하게 설명하고 3장에서는 본 논문에서 제안하는 알고리즘의 성능을 측정하기 위해 가정하는 시스템 환경에 대하여 소개한다. 4장에서는 PF 알고리즘의 압축 효율과 개선되는 계산 복잡도를 실험으로 보인다. 마지막으로 5장에서 결론을 맺는다.

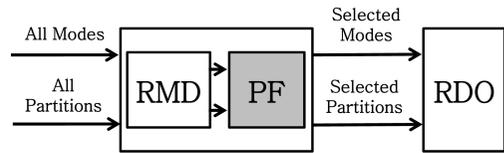


그림 1. 인트라 예측 전체 단계
Fig. 1 All stages for intra prediction

II. 제안하는 Partition Filtering 알고리즘

그림 1은 PF 알고리즘을 적용한 인트라 예측의 전체 단계를 나타낸 것이다. 기존의 HM과 같이 PF 알고리즘이 적용된 인트라 예측에서도 RMD 단계를 거쳐 35가지 예측 모드 중에서 수 개의 예측 모드 후보를 선택하게 된다. 하지만 PF 알고리즘이 적용된 인트라 예측에서는 RMD 단계가 끝나고 RDO를 수행하기 전에 PF 단계라는 새로운 단계를 추가하였다. PF 단계는 영상 특성을 고려하여 조합 가능한 모든 partition 중에서 해당 영역에 가장 적합한 몇 개의 partition 후보들을 선택하는 단계이다. PF 단계는 RMD 단계와 마찬가지로 RDO에서의 계산 복잡도 감소를 위해 추가되었으며, RMD 과정이 모두 끝나고 수행하게 된다. RMD와 PF 단계를 거쳐 선택된 수 개의 예측 모드와 수개의 partition 후보들만을 대상으로 인트라 예측 수행 후 최적의 모드와 최적의 블록 사이즈를 결정한다. 인트라 예측을 위해서는 주변 블록의 복원된 픽셀 데이터를 사용해야 한다. 이로 인해 한 블록의 예측을 위해서는 주변 블록의 예측, RDO 및 복원 과정이 모두 끝나야 하는 의존성 문제가 발생한다. 본 논문에서는 RMD와 PF 단계에서 다양한 partition 조합을 병렬적으로 빠르게 탐색하기 위하여 이웃 블록의 원영상 픽셀 데이터를 사용하여 예측함으로써 의존성을 제거하였다. RDO 단계에서는 블록 간 의존 관계를 지켜 복원 픽셀 데이터를 사용하여 최종 모드가 결정된다. 본 논문에서 가정하는 하드웨어 구조에서는 RMD와 PF 단계가 RDO 단계와 파이프라인으로 수행된다. 인트라 예측의 전체 단계에서 가장 복잡한 계산이 요구되는 RDO 단계가 critical path라고 볼 수 있다. 따라서 그림 1에서 selected mode와 partition의 개수를 줄이는 것은 인트라 예측 전체 단계의 수행 시간을 줄이는데 결정적인 역할을 하게 된다.

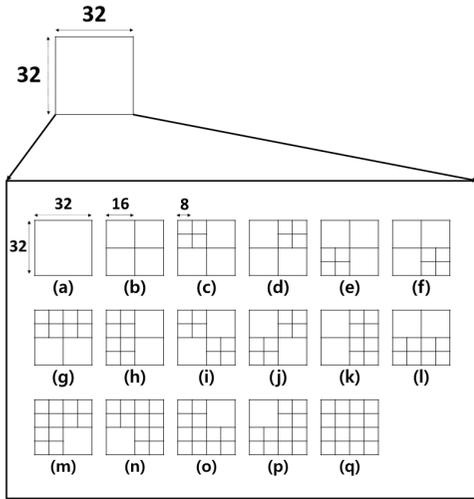


그림 2. Partition 조합
Fig. 2 Partition combination

그림 2는 32x32 크기를 갖는 Coding Tree Unit(CTU)에서 가능한 모든 partition 조합을 나타낸다. 그림 1에서 가장 상단에 존재하는 블록은 32x32 크기의 CTU를 나타낸다. CTU는 HEVC에서 사용하는 압축의 기본 단위로 32x32 크기의 CTU에서 partition들은 32x32 크기 블록과 16x16 크기 블록, 8x8 크기 블록의 조합으로 만들어진다. 예를 들어 partition (a)의 경우에는 32x32 크기 블록 1개로 구성된 partition이며, partition (d)의 경우는 16x16 크기 블록 3개와 8x8 크기 블록 4개로, partition (q)의 경우는 8x8 크기 블록 16개로 구성된 것이다. 이처럼 32x32 크기의 CTU에서는 총 17가지의 partition 조합이 가능하다.

그림 3에서는 본 논문에서 제안하는 PF 알고리즘에 대한 전체적인 흐름도를 나타내었다. CTU의 크기를 32x32로 설정하였을 때 Coding Unit(CU) 블록들의 크기는 32x32, 16x16, 8x8로 구성되며 각각 1개, 4개, 16개가 존재한다. RMD 단계를 거치면 각 블록 크기의 35가지 모드에 대한 SATD 값을 알 수 있다. Partition의 SATD 총합을 계산할 때, 각 블록의 SATD 값은 35가지 모드의 SATD 값 중에서 가장 작은 값을 사용하여 계산한다. 총 17가지의 partition에 대한 전체 SATD 값을 계산한 후 SATD 값을 기반으로 partition들을 오름차순으로 정렬한다. 이후 작은 SATD 값을 가지는 상위 N개의 partition을 선택한다. N개의 partition 중 32x32 크기 블록과 16x16 블록 4개로 구성된

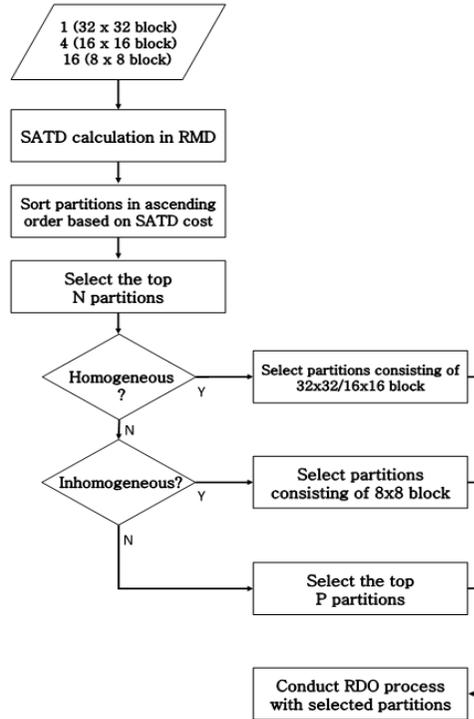


그림 3. PF 알고리즘 흐름도
Fig. 3 PF Algorithm flow chart

partition이 모두 포함되어 있을 경우 해당 영역은 균질적인 영상적 특성을 지닌다고 판단한다. 그리고 RDO 단계에 32x32 크기와 16x16 크기를 갖는 블록만을 수행하도록 넘겨주고 8x8 크기의 블록이 포함된 partition은 RDO를 수행하지 않는다. 만약 해당 영역이 균질적인 영상적 특성을 지니지 않는다고 판단되면 영상이 비균질적인 영상적 특징을 지니는지 다시 한 번 확인한다. 비균질적인 영상적 특성을 지니기 위한 조건은 다음과 같다. 먼저 상위 N개의 partition이 포함하는 블록들 중에서 그림 2의 partition (q)와 같이 8x8 크기의 블록 16개가 모두 선택되고, partition (a)와 같은 32x32 크기 블록이 선택되지 않아야 한다. 또한 partition (m)과 같이 16x16 크기의 블록이 4개 중 2개 포함되지 않을 때 해당 영역을 비균질적인 영상적 특성을 지닌다고 판단한다. 그리고 이 경우 partition (q)와 같이 8x8 크기의 블록으로만 구성된 partition에 대해서만 RDO 단계를 수행한다. 선택된 N개의 partition이 위에서의 균질적일 때의 조건과 비균질적일 때의 조건을 모두 만족하지 않는 경우, 이전

과정에서 오름차순 정렬한 partition 중에서 SATD 값이 작은 상위 P개의 partition을 다시 선택한다. 그리고 선택된 상위 P개의 partition에 포함된 블록들에 대해서만 RDO를 수행한다.

그림 3의 N과 P의 값은 영상 특성에 따라 다르게 주어지는 값이며 실험적으로 결정하였다. SATD와 같은 distortion 값을 이용하여 영상의 특성을 파악할 경우 작은 블록들로 조합된 partition들이 큰 블록들로 조합된 partition들보다 더 작은 SATD 값을 갖는 경향이 발생한다. 이로 인한 잘못된 partition 선택을 방지하기 위해 16개의 8x8 블록으로 이루어진 partition과 32x32 크기의 partition의 각각의 SATD 값을 비교한다. 8x8 블록으로 조합된 partition의 SATD 값을 1.2배 한 값이 32x32 블록 크기 partition의 SATD 값보다 크거나 같을 경우 그렇지 않은 경우보다 N과 P의 값을 크게 설정하였다. 이러한 이유는 영상 특성이 아주 명확하지 않다고 판단될 때 RDO 수행 단계에서 블록 선택의 폭을 넓혀주어 가장 적합한 partition 조합이 잘못 결정될 위험을 낮춰 주기 위함이다. 각각의 경우 N=14, P=7과 N=8, P=7로 값을 정하였다.

III. 시스템 환경

본 논문에서 제안하는 PF 알고리즘의 계산 복잡도 감소를 측정하기 위해 논문 [4]에서 제안하는 인트라 예측 인코더를 기반으로 RDO에서의 cycle 감소를 측정하였다. 논문 [4]에서 제안하는 인트라 예측 인코더는 하드웨어 모듈 간의 파이프라인 기법과 인터리빙을 활용한 블록 사이즈 수행 스케줄링을 통해 하나의 32x32 크기의 CTU가 800 cycle 안에 처리되도록 설계하였다. 논문 [4]에서 제공하는 인코더는 RMD 기법을 통해 RDO에서 수행하는 예측 모드의 수를 줄였다는 점에서는 PF 알고리즘과 동일하다. 하지만 PF 알고리즘에서는 예측 모드의 개수뿐만 아니라 영상 특성에 따른 Partition Filtering을 통해 RDO를 수행할 블록들의 개수 또한 줄여준다. 따라서 모든 크기의 블록에 대해 RDO를 수행하고 복원 픽셀을 만드는 논문 [4]에서의 인코더와 비교하면 PF 알고리즘은 논문 [4]에서 소개하는 인코더와 알고리즘 면에서는 큰 차이가 있다. 그렇기 때문에 만약 PF 알고리즘의 성능을 논문 [4]에서 제안하는 스케줄링을 기반으로 그대로 활용한다면 블록 간의 의존성 문제가 발생하여 올바른 성능 측정 방법이 될 수 없다.

표 1. 단계 별 계산량
Table 1. Numbers of cycles for processing stages

Block Size	Candidate Mode Number	Cycles Per Block			
		Intra Predictor	DCT	RD/MD	IDCT/REC ON
32 x 32	3	16	64	64	128
16 x 16	3	4	8	8	2
8 x 8	8	1	1	1	1

이에 따라 본 논문에서는 PF 알고리즘의 성능을 측정하기 위해 새로운 스케줄링 방법을 도입하였다. 먼저 PF 알고리즘이 적용된 인트라 예측 인코더에서 하나의 CTU를 처리하기 위해 필요한 단계는 논문 [4]에서 제안하는 인코더와 마찬가지로 intra prediction 단계, DCT 단계, rate-distortion 단계, mode-decision 단계, IDCT 단계, reconstruction 단계로 구성되어 있다. 표 1은 논문 [4]의 인코더에서 각 모듈마다 소요되는 cycle량을 보여준다. 표 1의 첫 번째 열은 CU 블록의 크기를, 두 번째 열은 RMD 단계를 거쳐 선택된 예측 모드 후보의 개수를 나타낸다. 세 번째에서 여섯 번째 열들에서는 각각의 모듈에서 CU 블록 크기에 따라 소요되는 cycle의 양을 보여준다. 32x32 사이즈의 블록은 주변 블록과의 의존성이 없다. 따라서 16x16 크기 블록과 8x8 크기 블록을 주변 블록과의 의존성을 고려하여 순차적으로 수행하는 단계를 인트라 인코딩의 critical path로 설정하고, 32x32 크기 블록의 처리는 16x16 크기 블록과 8x8 크기 블록 사이에 인터리빙 방식으로 삽입되어 처리하도록 스케줄링 하였다.

그림 4는 인트라 예측 및 RDO를 통해 최적 모드를 결정하기 위한 블록 별 스케줄링을 나타낸다. 그림 4(a)는 모든 크기의 CU 블록에 대해 인트라 예측과 RDO를 수행하는 기존의 방법을 기반으로 작성한 스케줄링이다. 이때 32x32 크기 블록의 처리는 인터리빙 방식으로 삽입되며 이에 대한 표기는 하지 않았다. 그림 4(a)의 왼쪽에는 표 1에서 보인 각 단계가 나타나 있으며 하나의 CU 블록이 수행되는 순서대로 표기되어 있다. 여러 가지 크기의 블록들과 아래의 숫자들은 CU 블록 크기에 따른 각각의 단계에서 소요되는 cycle량을 나타낸 것이다. 그림 4(b)는 PF 알고리즘을 통해 32x32 크기 블록과 16x16 크기 블록이 필터링이 되고, 8x8 크기 블록에 대해서만 인트라 예측과 RDO를 수행한다고 가정하였을 경우 변경되는 스케줄링을 나타낸다. 또 다른 예로 그림 4(c)는 PF 알고리즘을 통해

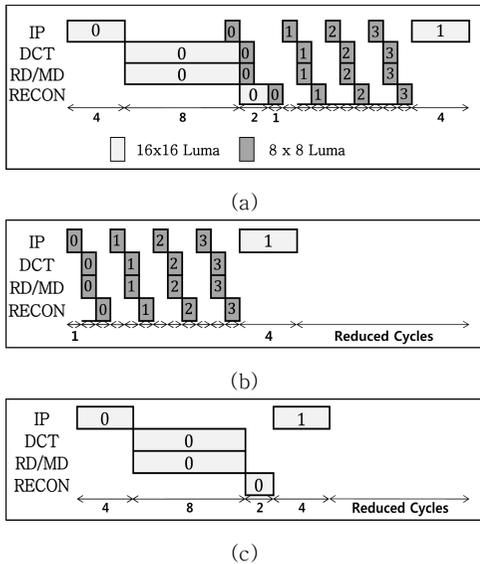


그림 4. 스케줄링 (a) 기존 방법 (b) PF 알고리즘 예시 1 (c) PF 알고리즘 예시 2
 Fig. 4 Scheduling (a) Original Method (b) PF Algorithm Example 1 (c) PF Algorithm Example 2

8x8 크기 블록이 필터링이 되어, 32x32 크기 블록과 16x16 크기 블록에 대해서만 인트라 예측과 RDO를 수행하는 경우의 스케줄링을 보여준다. 그림 4(a)에서는 16x16 크기 블록 1개와 8x8 크기 블록 4개를 처리하는데 총 24cycle이 소요된다. 또한 인터리빙 방식으로 추가된 32x32 크기 블록을 처리하기 위한 cycle까지 추가되면 총 cycle은 24cycle을 훨씬 웃돌 것이다. 하지만 PF 알고리즘이 적용된 그림 4(b)의 예시를 보면, 소요되는 cycle 양은 12 cycle로 기존의 방법에 비해 cycle 양이 크게 감소한다는 것을 알 수 있다.

PF 알고리즘을 적용한 인코더는 후보 partition으로 32x32 크기 CU 블록이 선택되었을 때보다 그렇지 않을 경우에 더 큰 cycle 감소량을 보인다. 왜냐하면 32x32 크기 블록을 처리하는데 필요한 총 cycle 양은 80 cycle로 다른 크기의 블록들에 비해 압도적으로 많기 때문에 만약 PF 알고리즘이 적용되어 다른 크기의 블록들이 모두 생략된다고 할지라도 적어도 80 cycle은 소요될 것이기 때문이다. 따라서 위에서 소개한 스케줄링은 단순히 기존의 인코더와 비교하여 감소할 수 있는 cycle의 양을 이해하기 쉽도록 도식화한 것이다. 실제 PF 알고리즘을 적용한 하드웨어 기반 인코더를 작성할

때에는 각각의 CTU마다 소요되는 cycle의 양이 일정하게 유지하도록 하는 추가적인 기법이 필요할 수 있다.

IV. 실험 결과

본 논문에서 제안하는 알고리즘의 성능을 실험을 통해 알아보려고 한다. 계산 복잡도는 다음과 같은 방법으로 측정한다. 먼저 모든 크기의 블록들에 대해 RDO를 수행할 때 소요되는 RDO 모듈에서의 cycle 양을 측정한다. 다음 PF 알고리즘을 적용하여 몇 개의 선택된 크기의 블록에 대해서만 RDO를 수행할 경우 소요되는 RDO 모듈에서의 cycle 양을 측정한다. 이후 두 방법 간의 RDO 모듈에서의 전체 cycle 양을 비교하고, 감소하는 cycle의 양을 기반으로 계산 복잡도 감소량을 측정한다. 먼저 영상 특성이 잘 나타나는 영상들을 이용하여 PF 알고리즘의 효과를 파악해 보고자 한다. 균질적인 특성이 잘 나타나는 영상으로 B Class의 ParkScene과 E Class의 KristenAndSara를 선택하고, 비균질적인 특성이 잘 나타나는 영상으로는 A Class의 PeopleOnStreet와 C Class의 BQMall을 선택하였다. HEVC reference software로 HM 13.0을 사용하였으며 영상마다 각각 100 프레임씩 인코딩하여 실험을 진행하였다. 양자화 파라미터는 22, 27, 32, 37로 설정하였으며 전체 cycle은 각각의 파라미터에 따라 소요된 cycle의 평균을 사용하였다. 그림 5는 앞에서 선택한 4개의 영상에 대하여 기존의 HM 13.0에서 제공하는 방법으로 인코딩을 하였을 때와 PF 알고리즘을 적용하였을 때의 각각의 RDO 수행 단계에서 소요되는 총 cycle의 양을 보여준다. 그림의 가로축 영상 1, 2, 3, 4는 각각 ParkScene, KristenAndSara, PeopleOnStreet와 BQMall을 의미하며 세로축은 HM 13.0에서 소요된 cycle 양을 100으로 환산했을 때 각각의 영상에서 소요된 cycle의 비율을 나타낸다. 그림 5를 분석하면 균질한 특성이 잘 나타나는 영상에 비해 비균질적인 특성이 잘 나타나는 영상에서 감소하는 총 cycle의 양이 더 크다는 것을 알 수 있다. 따라서 이 결과를 통해 PF 알고리즘이 적용된 인코더는 균질적인 영상 특성을 많이 포함하는 영상보다 비균질적인 영상 특성을 많이 포함하는 영상일수록 계산 복잡도 감소 효과가 크다는 것을 알 수 있다.

다음은 기존의 방법과 비교하여 PF 알고리즘의 압축 성능과 RDO에서의 전체 cycle 양 감소 정도

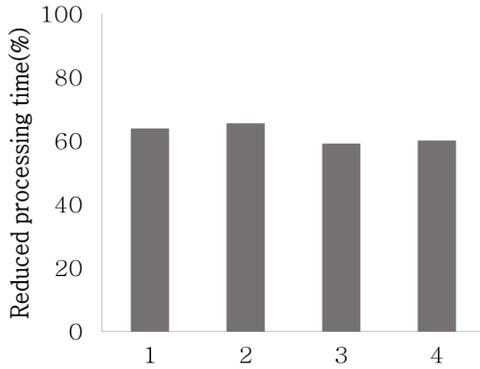


그림 5. 영상 특성에 따른 Cycle 감소 비교
Fig. 5 Cycle comparison for image characteristic

를 알아보기 위한 실험을 진행하였다. 실험을 위해 소프트웨어는 HM 13.0을 사용하였고 100 프레임씩 인코딩한 결과를 사용하였다. 양자화 파라미터는 22, 27, 32, 37로 설정하였다. 실험 영상은 Class A에서 Traffic, PeopleOnStreet, Class B에서 BQTerrace, Cactus, ParkScene, Class C에서 BQMall, BasketballDrill, PartyScene, Class D에서는 BasketballPass, BlowingBubbles, BQSquare, Class E에서 FourPeople, KristenAndSara를 사용하였다. 표 2는 제안하는 방법을 적용하였을 때 Bjontegaard-Delta Bit-Rate(BDBR)의 증가와 계산 복잡도의 감소량을 나타낸다. 표의 첫 번째 열과 두 번째 열은 영상의 Class와 Sequence를 나타낸다. 세 번째 열은 BD RATE 값을 % 단위로 나타낸 것이고, 네 번째 열은 BD PSNR 값을 dB 단위로 나타내었다. 다섯 번째 열의 CYCLE은 수식 (1)과 같은 방법으로 계산되었다.

Complexity reduction

$$= \frac{\text{기존}cycle - PF\ cycle}{\text{기존}cycle} * 100 (\%) \quad (1)$$

표의 마지막 행에서는 모든 영상에 대한 BD RATE, BD PSNR, CYCLE에 대한 평균값을 나타내었다. 본 논문에서 제안한 PF 알고리즘을 적용하였을 때 BD RATE의 평균적으로 0.95% 증가하였고, BD PSNR은 평균적으로 -0.0119[dB]만큼을 감소하였음을 확인할 수 있었다. 이에 반해 RDO에서의 cycle 감소는 평균적으로 37.64%로 계산 복잡도는 크게 낮아졌음을 알 수 있다.

표 2. 압축 효율 감소와 계산 복잡도 감소량
Table 2. Losses in the compression efficiency and reduced calculation complexity

Class	Sequence	BD RATE [%]	BD PSNR [dB]	Complexity Reduction [%]
A	Traffic	1.08	-0.0009	39.85
	PeopleOnStreet	0.82	-0.0124	40.75
B	BQTerrace	1.14	-0.0372	35.16
	Cactus	1.15	0.0028	37.72
	ParkScene	0.92	0.0054	35.97
C	BQMall	1.39	-0.0062	39.77
	BasketballDrill	0.86	-0.0297	39.50
	PartyScene	0.08	-0.0026	38.44
D	BasketballPass	1.12	-0.0306	38.88
	BlowingBubbles	0.19	-0.0001	39.64
	BQSquare	0.13	-0.0039	38.96
E	FourPeople	1.02	-0.0141	37.89
	KristenAndSara	2.49	-0.0254	34.40
Average		0.95	-0.0119	38.23

V. 결론

본 논문에서는 영상 특성을 파악하여 적합한 블록 크기를 미리 결정해줌으로서 RDO에서의 계산 복잡도를 감소시키는 방법에 대하여 분석하고 연구하였다. 하다마드 변환을 통해 얻어지는 SATD 값을 이용하여 영상 특성을 파악하여 영상을 균질적인 특성을 지니는 경우와 비균질적인 특성을 지니는 경우, 해당 영역의 영상적 특성을 판단할 수 없는 경우로 세 가지로 나누고 주어진 조건에 따른 Partition Filtering을 진행하였다. 그리고 RDO를 수행할 후보 partition들을 선정하여 이들이 지닌 CU 블록들에 대해서만 RDO 단계를 수행해하였다. 실험 결과 압축 성능의 손실이 거의 없이 평균 38.23%만큼의 계산 복잡도를 감소시킬 수 있었다. 근래 반도체, 통신 기술이 발전함에 따라 고해상도 영상에 대한 수요가 점점 많아지고 있으며 모바일과 임베디드 시스템에서 사용될 고속 인코더 개발을 위한 연구가 다양하게 진행되고 있다. 논문에서 제안하는 알고리즘에 대한 실험 결과는 고해상도 영상을 실시간으로 처리하는데 필요한 인코더를 개발하는데 적합한 알고리즘이 될 수 있음을 보여준다. 향후 영상 특성을 파악하는 방법과 함께 RDO에서의 계산 복잡도를 낮출 수 있는 다양한 알고리즘을 연구할 계획이다.

References

[1] J. Zhu, Z. Liu, D. Wang, Q. Han, Y. Song,

- “Fast prediction mode decision with Hadamard transform based rate-distortion cost estimation for HEVC intra coding,” Proceedings of IEEE International Conference on Image Processing, pp. 1977-1981, 2013.
- [2] W. Zhao, T. Onoye, T. Song, “hardware-oriented fast mode decision algorithm for intra prediction in HEVC,” Proceedings of Picture Coding Symposium, pp. 109-112, 2013.
- [3] Y. Zhao, L. Song, X. Wang, M. Chen, J. Wang, “Efficient Realization of Parallel HEVC Intra Encoding,” Proceedings of IEEE International Conference on Multimedia and Expo workshops, pp 1-6, 2013.
- [4] G. Pastuszak, A. Abramowski, “Algorithm and Architecture Design of the H.265/HEVC Intra Encoder,” IEEE Transactions on Circuit and System for Video Technology, Vol. 26, pp. 210-222, 2016.

Seung-Won Park (박승원)



In 2009, he joined the Department of Computer Science and Information Engineering at Inha University, Korea. His research interests include algorithm and architecture design of video coding for HEVC and H.264/AVC and configurable video coding for real time systems.

Email: koporia@gmail.com

Chae Eun Rhee (이채은)



She received the B.S., M.S. and Ph.D. degrees in Electrical Engineering and Computer Science from Seoul National University, Seoul, Korea, in 2000, 2002 and 2011, respectively. From 2002 to 2005, she was with the Digital TV Development Group, Samsung Electronics Company Ltd., Suwon City, Korea, as an Engineer, where she was involved in bus architecture and MPEG decoder development. In 2013, she joined the Department of Information and Communication Engineering at Inha University, Korea, where she is currently working as an assistant professor. Her research interests include algorithm and architecture design of video coding for HEVC and H.264/AVC and configurable video coding for real time systems.

Email: chae.rhee@inha.ac.kr