

특집논문 (Special Paper)

방송공학회논문지 제22권 제2호, 2017년 3월 (JBE Vol. 22, No. 2, March 2017)

<https://doi.org/10.5909/JBE.2017.22.2.193>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

HEVC 실시간 소프트웨어 인코더에서 GOP 병렬 부호화를 지원하는 R-lambda 모델 기반의 율 제어 방법

김대은^{a)}, 장용준^{a)}, 김문철^{a)‡}, 임웅^{b)}, 김휘용^{b)}, 석진욱^{b)}

R-lambda Model based Rate Control for GOP Parallel Coding in A Real-Time HEVC Software Encoder

Dae-Eun Kim^{a)}, Yongjun Chang^{a)}, Munchurl Kim^{a)‡}, Woong Lim^{b)}, Hui Yong Kim^{b)},
and Jin Wook Seok^{b)}

요약

본 논문에서는 4K UHD 입력 영상을 실시간으로 부호화하기 위해 GOP 단위 또는 IDR 주기 단위의 병렬 부호화 구조를 지원하는 R- λ 모델 기반의 율 제어 방법을 제안한다. 제안하는 R- λ 모델 기반의 율 제어 방법에서는 순차적 프레임 부호화가 아닌 병렬 부호화를 위한 슬라이스 레벨 비트 분배(bit allocation) 방법을 제안한다. GOP 단위 또는 IDR 주기 단위의 병렬 부호화에서 율 제어를 작동시키는 경우, 계층적 B 구조에서 최하위 프레임 계층을 제외한 동일 계층에 속한 프레임 간에는 상호간에는 부호화 수행 이후 각 소모된 비트양에 대한 정보를 공유할 수 없기 때문에 기존의 비트량 분배 방식으로는 비트 예산(bit budget) 관리가 불가능하다. 이를 해결하기 위해 본 논문에서는, 기존의 R- λ 모델 기반 율 제어 방법인 프레임 부호화 순서에 따라 각 프레임별로 목표 비트량 분배하던 방식으로부터, GOP 별로 비트량을 할당한 후, 각 GOP 내의 계층적 B 구조에서 계층이 깊어지는 방향으로 순차적으로 비트 예산을 갱신하여 비트량을 분배하는 방식으로 율 배분 방식을 개선하였다. 뿐만 아니라, 입력 영상의 전처리 과정을 통해 획득된 영상의 복잡도 정보를 고려하여 비트를 분배하여 영상의 주관적인 화질을 향상시켰다. 실험을 통해 제안 방법이 병렬 구조의 HEVC 부호화기에서 잘 작동함을 확인 할 수 있었고, 전처리 결과를 활용하여 율 제어기의 성능을 향상시킬 수 있음을 확인하였다.

Abstract

In this paper, we propose a rate control method based on the R- λ model that supports a parallel encoding structure in GOP levels or IDR period levels for 4K UHD input video in real-time. For this, a slice-level bit allocation method is proposed for parallel encoding instead of sequential encoding. When a rate control algorithm is applied in the GOP level or IDR period level parallelism, the information of how many bits are consumed cannot be shared among the frames belonging to a same frame level except the lowest frame level of the hierarchical B structure. Therefore, it is impossible to manage the bit budget with the existing bit allocation method. In order to solve this problem, we improve the bit allocation procedure of the conventional ones that allocate target bits sequentially according to the encoding order. That is, the proposed bit allocation strategy is to assign the target bits in GOPs first, then to distribute the assigned target bits from the lowest depth level to the highest depth level of the HEVC hierarchical B structure within each GOP. In addition, we proposed a processing method that is used to improve subjective image qualities by allocating the bits according to the coding complexities of the frames. Experimental results show that the proposed bit allocation method works well for frame-level parallel HEVC software encoders and it is confirmed that the performance of our rate controller can be improved with a more elaborate bit allocation strategy by using the preprocessing results.

Keyword : HEVC, Rate control, R-lambda model, GOP parallelism, real time encoder

Copyright © 2017 Korean Institute of Broadcast and Media Engineers. All rights reserved.

“This is an Open-Access article distributed under the terms of the Creative Commons BY-NC-ND (<http://creativecommons.org/licenses/by-nc-nd/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited and not altered.”

1. 서론

가장 근래에 우수한 압축 성능을 발휘하는 것으로 알려져 있는 HEVC (High Efficiency Video Coding) 비디오 부호화 표준^{[1][2]}은 기존의 H.264 AVC (Advanced Video Coding)^{[3][4]}와 비교하여 주관적 화질이 동일한 수준일 때 약 두 배 정도의 부호화 효율을 달성하였다^[2]. 이같이 높은 압축 효율을 달성하기 위해서 HEVC 부호화기에는 기존의 다른 비디오 표준 부호화기에 비해 다양한 부호화 도구가 개발되어 적용되어 있다. 이에 따라 부호화 과정에 필요한 계산 복잡도 역시 크게 증가하였고, 따라서 처리 시간 또한 크게 증가하게 되었다. 더욱이 양질의 비디오 콘텐츠에 대한 수요가 증가함에 따라 1920×1080의 full HD비디오뿐만 아니라 3840×2160 4K UHD(Ultra HD)의 해상도를 갖는 초고해상도 영상이 보편화되어 처리해야 하는 데이터의 양이 크게 증가되었다. 이같이 부호화기 자체의 복잡도 증가와 영상의 해상도 증가로 인해 부호화기의 부호화 시간을 단축시키는 것은 이전보다 더욱 어려운 문제가 되었다. 이같은 문제를 해결하기 위해 다양한 방법이 시도되고 있으며^{[5][7]}, 그 중 한 방법으로 컴퓨터 시스템의 다중코어를 이용한 병렬 부호화 처리 방법이 제안되기도 하였다^[8]. [8]에서는 계층적 B 픽처의 참조 관계를 제거함으로써 독립적으로 처리될 수 있는 프레임의 수를 늘려 이들을 동시에 병렬적으로 처리하여 부호화 속도를 증가시켰다. 더 자세한 내용은 본 논문의 II장에서 다시 한 번 설명하도록 한다.

한편 방송 시스템이나 인터넷 스트리밍 서비스와 같이 고정된 대역폭 조건에서 안정적으로 비트스트림을 전송하

기 위해서는 비트 율 제어가 부호화기에 반드시 필요하다. 율 제어기는 일반적으로 비트 율 모델과 비트 예산(bit budget) 관리로 구성되어 있다. 응용 분야에 따라 사용자가 원하는 비트 율로부터 각 프레임에 대해 얼마만큼의 비트를 할당할 것인지를 결정하고, 이미 부호화된 프레임에서 목표 비트와 실제 출력 비트 사이에 발생한 오차를 관리하여 평균적인 관점에서 목표 비트 율에 근접하는 비트 율을 출력하도록 하는 것이 비트 예산 관리 부분이고, 각 프레임에 대해 목표 비트가 할당되었을 때, 해당 프레임에 대해 부호화기가 목표 비트만큼을 출력 할 수 있도록 적절한 양자화 파라미터(QP, quantization parameter)를 계산 하는 과정은 율 모델에 의해 결정된다. 대부분의 비트 율 제어 연구는^{[9][12]} 여러 프레임이 병렬적으로 동시에 부호화 되는 경우가 아닌, 한 프레임씩 부호화 되는 경우만을 상정하고 연구가 진행되어 왔다. 하지만, 앞서 언급한 바와 같이 비디오 부호화기의 처리 속도를 높이기 위해서는 병렬 부호화가 불가피하다. 기존의 한 프레임씩 부호화 되는 경우만을 가정한 연구들에서 제안된 비트 예산 관리 방법은 프레임 수준의 병렬 처리 방식이 적용되는 부호화기에 바로 적용하기에는 부적합하다. 더욱이 프레임 수준의 병렬 처리에 더하여 부호화 속도를 높이기 위해, GOP(Group of Picture) 및 IDR(Instantaneous Decoder Refresh) 주기 수준의 병렬 부호화 처리가 설계된, 여러 프레임을 동시에 부호화하는 부호화기에서는 안정적인 출력 비트 율 제어가 어렵다.

이 문제를 해결하기 위해, 본 논문에서는 HEVC 부호화 표준의 참조 소프트웨어인 HM^[13]에 구현되어 있는 R-λ 모델 기반의 율 제어 방법^{[9][10]}에서 제안된 율 제어 구조를 개선하여 GOP 및 IDR 주기 단위의 병렬 부호화 구조에서도 안정적으로 출력 비트 율을 제어할 수 있는 방법을 제안한다. 이와 더불어, 비트 율 제어 정확도는 물론 부호화 성능을 더 높이기 위해 전 처리 결과를 활용하는 율 분배 방식을 제안한다. 제안 방법의 타당성 및 효과를 검증하기 위해 GOP 및 IDR 주기 단위의 병렬 부호화 구조가 적용된 소프트웨어 HEVC 부호화기에 적용하여 4K UHD 영상에 대해 실험하였다.

본 논문의 구성은 다음과 같다. II장에서는 HEVC의 R-λ 모델 기반 율 제어 방법을 설명하고, III장에서는 GOP 및 IDR 주기 단위의 병렬 부호화기의 구조에 대해 설명한다. IV장에서는 병렬 구조의 부호화기를 지원하기 위해 본 연

a) 한국과학기술원 전기및전자공학부(The School of Electrical Engineering, Korea Advanced Institute of Science and Technology)

b) 한국전자통신연구원 방송·미디어연구소 미디어연구본부 실감AV연구그룹 (Realistic AV Research Group Media Research Division Broadcasting·Media Research Laboratory Electronics and Telecommunications Research Institute)

✉ Corresponding Author : 김문철(Munchurl Kim)

E-mail: mkim@kaist.ac.kr

Tel: +82-42-350-7419

ORCID: <http://orcid.org/0000-0003-0146-5419>

※ 이 논문은 2016년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (B0101-16-0295, 초고품질 콘텐츠 지원 UHD 실감방송/디지털 시네마/사이니지 융합 서비스 기술 개발).

Manuscript received January 13, 2017; Revised March 3, 2017; Accepted March 3, 2017.

구에서 제안하는 비트 분배 방법을 설명한다. V장에서는 이에 대한 실험결과를 제시하고, 마지막으로 VI장에서 이 논문의 결론을 짓는다.

II. HEVC의 R-λ 모델 기반 율 제어 방법

HEVC 표준의 참조 소프트웨어인 HM 인코더에 채택된 R-λ 모델 기반의 율 제어 알고리즘은 율 모델과 비트 분배 방법으로 구성되어 있다. 본 장에서는 각 부분에 대하여 자세히 설명한다.

1. R-λ 모델

율 제어에 있어서 율 모델은 매우 중요하다. 원하는 비트를 출력하기 위해 양자화 파라미터 (quantization parameter, QP) 값을 어떻게 결정할 것인가에 대한 모델이 바로 율 모델이기 때문이다. R-λ 모델이 등장하기 이전의 율 모델은 QP로부터 비트를 예측하는 R-Q 모델을 사용하였다. 그러나 [10]에서 HEVC 표준의 경우 기존의 비디오 부호화 표준에 비해서, 부호화 매개변수들의 가능한 조합이 상당히 많아졌고, 이를 단순히 QP에 관한 함수로만 표현하기보다는 λ에 관한 함수로 표현하는 것이 더 바람직하다는 관찰을 바탕으로 R-λ 모델을 제시하였다. [10]의 율 제어 알고리즘은 출력 비트 양과 λ와의 관계를 아래 식과 같이 모델링 하여 이용한다.

$$\lambda = \alpha \cdot bpp^\beta \quad (1)$$

여기서 α와 β는 모델 파라미터이고, bpp는 화소당 비트 수로서 한 프레임에서 발생하는 비트 수를 한 프레임의 화소 수로 나눈 값이다. 모델 파라미터 α와 β는 영상의 특성에 따라서 달라지는데 [10]에서는 초기 값으로 α=3.2003와 β=-1.367을 사용하였다. 그리고 영상이 부호화됨에 따라 모델 파라미터를 아래의 식을 통해 업데이트 해가면서 모델의 정확도를 높였다^[10].

$$\lambda_{comp} = \alpha_{old} \cdot bpp_{real}^{\beta_{old}} \quad (2)$$

$$\alpha_{update} = \alpha_{old} + \delta_\alpha \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \alpha_{old} \quad (3)$$

$$\beta_{update} = \beta_{old} + \delta_\beta \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \ln bpp_{real} \quad (4)$$

여기서 δ_α = 0.1, δ_β = 0.05 이고, b_{pp_{real}}은 부호화 후에 실제로 출력된 비트를 의미한다. 율 제어를 하고자 하는 프레임의 목표 비트가 b_{pp}(bit per pixel)로 주어지면 모델 파라미터에 의해 λ값이 계산되고 이 λ값은 QP를 결정하는데 사용되며, λ와 QP의 관계는 아래의 식과 같다^[10].

$$QP = 4.2005 \ln \lambda + 13.7122 \quad (5)$$

(5)를 통해 결정된 QP 값이 사용되어 부호화 되고, R-λ 모델에 의거하여 해당 프레임에서는 b_{pp}만큼의 비트가 출력될 것으로 기대할 수 있는 것이다.

2. 비트 분배 방법

응용 분야에 따라 시퀀스 전체에 대한 목표 비트 율이 정해지면, 정해진 목표 비트 율과 현재 버퍼 상태를 고려하여 GOP에 대한 목표 비트 T_{GOP}가 할당된다.

$$T_{GOP} = \left(R_{pcAvg} + \frac{R_{pcAvg} \cdot N_{coded} - R_{coded}}{SW} \right) \cdot N_{GOP} \quad (6)$$

여기서 R_{pcAvg}는 시퀀스 전체에 대한 목표 비트 율을 단순히 프레임 율로 나눈, 프레임당 평균 목표 비트이고, N_{coded}는 부호화 된 프레임의 수, R_{coded}는 부호화 된 비트 수, SW는 스무딩 윈도우(smooth window)이고, N_{GOP}는 GOP의 크기이다. R_{pcAvg} · N_{coded}는 프레임당 평균 목표 비트에 GOP의 크기를 곱한 값으로 현재 버퍼 상태를 고려하지 않았을 때의 현재 GOP에 할당하는 비트이다. 그러나 현재 버퍼 상태를 고려하면 (6)의 분수식이 더해지는데, 분수식의 분자는 평균적으로 소모되었어야 할 비트에서 실제 발생한 비트의 차이를 계산한 식이고 분모의 SW는 발생한 오차를 얼마만큼 반영할 것인가 하는 인수이다. GOP에 대한 목표 비트가 계산된 다음에는 GOP의 구조와 현재 프레임의 계층, 전체 목표 비트율을 고려하여 현재 프레임의 목표 비트 T_{CurrPc}가 계산된다.

$$T_{CurrPc} = \frac{T_{GOP} - Coded_{GOP}}{\sum_{UncodedPcs} \omega_i} \cdot \omega_{CurrPc} \quad (7)$$

여기서 $Coded_{GOP}$ 는 현재 프레임이 포함된 GOP에서 현재 프레임을 부호화하기 이전에 부호화된 프레임에서 출력된 비트 수이고, ω_i 는 GOP의 구조에 따라 각 프레임마다 정해지는 중요도이다. 표 1은 ω 를 나타낸다.

표 1. 목표 비트와 프레임 계층에 따른 ω

Table 1. ω depending on target bpp and frame level

Level	bpp>0.2	0.2≥bpp>0.1	0.1≥bpp>0.05	other wise
1	15	20	25	30
2	5	6	7	8
3	4	4	4	4
4	1	1	1	1

이같이 기존의 목표 비트 분배 방법은 한 프레임씩 부호화를 진행해 감에 따라 목표 비트율과 현재까지의 실제 출력 비트 율을 비교하여 현재 프레임의 목표 비트를 조정해 가며 최종 목표 비트 율을 달성하였다. 그러나, 1장에서 언급한 바와 같이 GOP 레벨 및 IDR 주기 레벨로 병렬 부호화가 적용된 경우, 여러 프레임이 동시에 부호화되기 시작하기 때문에, 현재 프레임에 비트를 분배하기 위한 새로운 방법이 필요하다. GOP 레벨 및 IDR 레벨의 병렬화에 대해서는 III장에서 자세히 설명한다.

III. GOP 및 IDR 주기 단위의 병렬 부호화 구조

HEVC 부호화기는 기존의 부호화기에 비해 다양한 부호화 도구와 조합을 지원하기 때문에 연산 복잡도가 매우 증

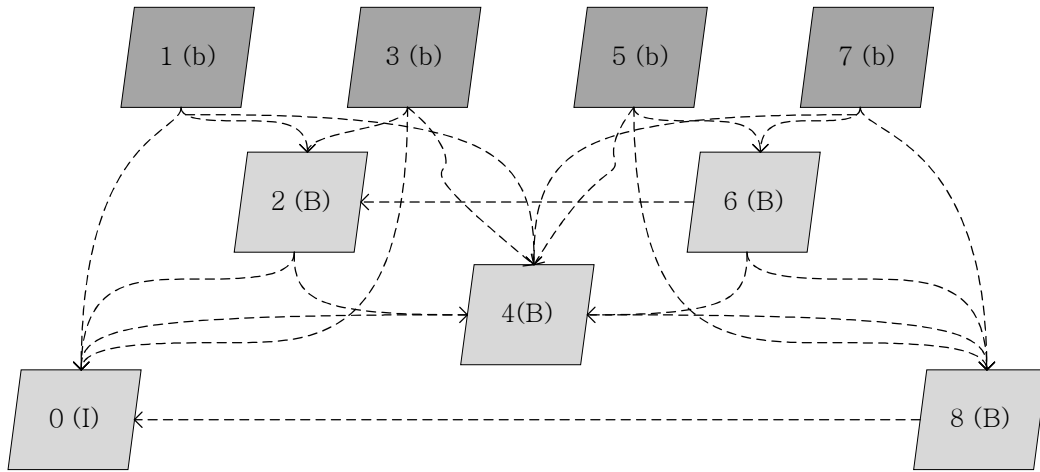


그림 1. HEVC 임의 접근 GOP 구조

Fig. 1. GOP structure of HEVC random access

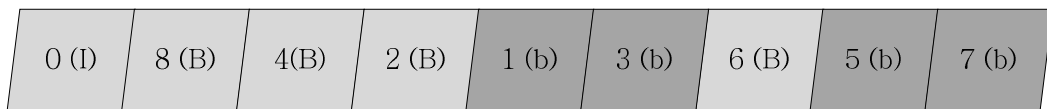


그림 2. 임의 접근 GOP의 부호화 순서

Fig. 2. Encoding order of random access GOP

가한다. 이러한 HEVC 부호화기의 부호화 속도를 증가시키고 나아가서 실시간 부호화기를 구현하기 위해서는 소프트웨어 부호화기에서의 병렬적 부호화 구조 설계가 필수적이다. 본 장에서는 [8]에서 제안된 프레임 레벨 HEVC 병렬 부호화에 대해 설명한다. 그림 1은 HEVC의 일반적인 임의 접근 (random access) GOP 구조를 나타낸다.

그림 1에서와 같이 I 픽처를 제외한 B 픽처와 b 픽처는 화살표가 가리키는 프레임을 참조함으로써 부호화 효율이 향상되지만, 이에 따르는 프레임간의 의존성(dependency)이 발생하여 프레임을 병렬적으로 부호화 하는데 제한이 발생한다. 그림 2는 일반적인 순차 처리 부호화기에서 그림

1과 같은 구조의 GOP를 부호화 할 때, 부호화 되는 프레임의 순서를 POC 로 나타낸다.

그림 2에서 6번 프레임은 2번 프레임에 대해 의존성을 가지므로, 서로 독립적으로 부호화될 수 있는 프레임은 1, 3, 5, 7 번 프레임뿐이고 병렬 구조를 적용할 수 있는 프레임도 이들 뿐이다. 이에, [8]논문에서는 6번 프레임의 2번 프레임에 참조를 끊는 GOP구조를 제안하였다. 그림 3은 [8]에서 제안한 프레임 수준 병렬화를 위한 GOP 구조를 나타낸다.

그림 3에서 볼 수 있듯이 6번 프레임이 더 이상 2번 프레임 참조하지 않으므로 2번 프레임과 6번 프레임은 동시

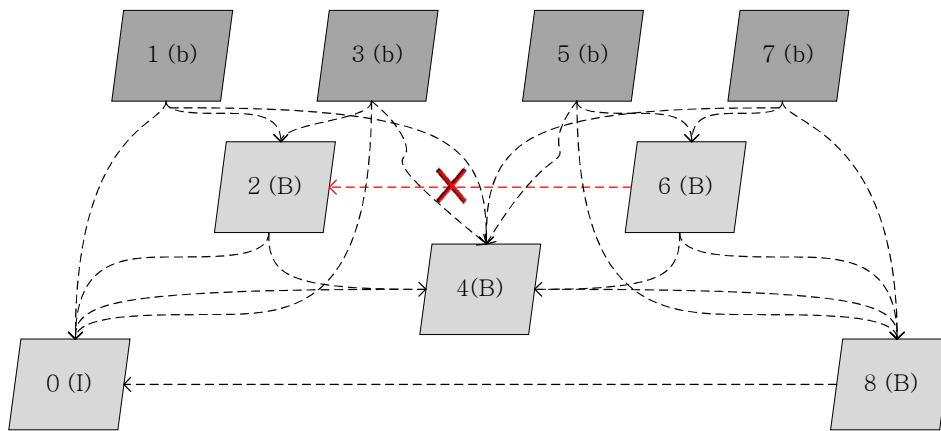


그림 3. 프레임 수준의 병렬화를 위한 GOP 구조
 Fig. 3. GOP structure for frame level parallelism

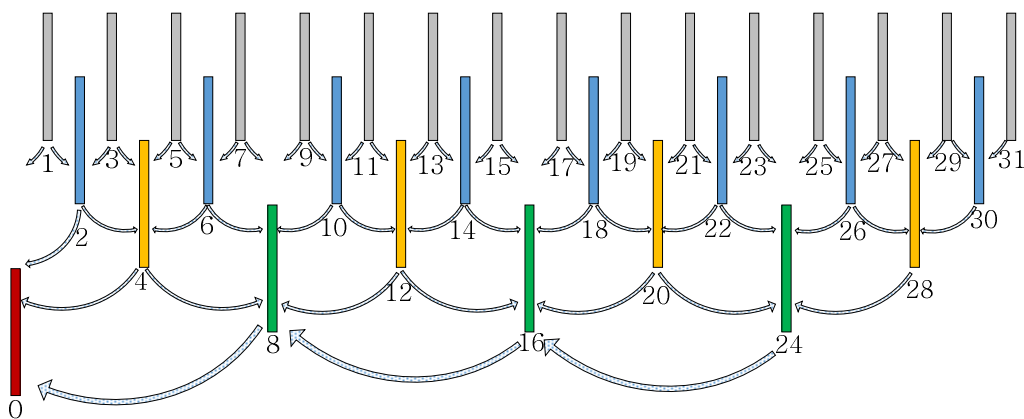


그림 4. 프레임 수준의 병렬화를 위한 GOP 구조
 Fig. 4. GOP structure for frame level parallelism

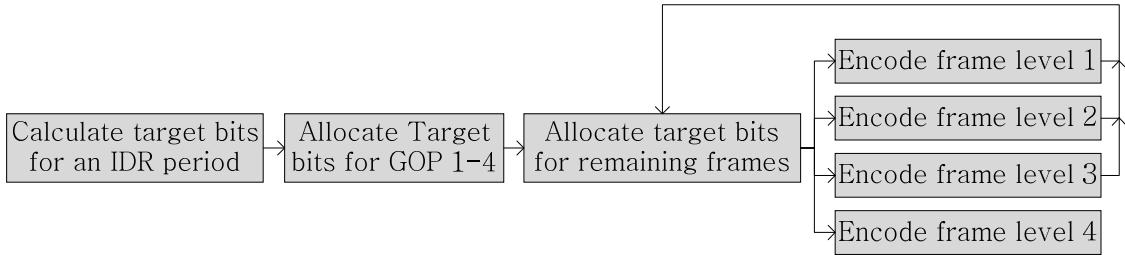


그림 6. 제안 방법의 율 분배 순서도
Fig. 6. Flow chart for proposed bit allocation method

1. 율 분배 방법

율 제어 알고리즘은 율 모델과 비트 율 분배로 구성되어 있는데, 현실적으로 율 모델의 정확도에는 한계가 있기 때문에, 현재 부호화 하려는 프레임에 할당된 목표 비트에 대한 실제 출력 비트의 오차가 발생할 수밖에 없다. 일반적인 순차적 프레임 부호화기에서는 발생하는 오차를 그 다음 프레임의 율 분배 과정에서 고려하여 부호화가 진행되어감에 따라 발생하는 오차를 해결하도록 하였다. 그러나 III장에서 언급한 병렬화 부호화기에서는 동시에 여러 프레임의 부호화가 시작되기 때문에 순차적 부호화기에서 사용되는 비트 분배 방법을 그대로 적용할 수 없다. 이에, 본 논문에서는 병렬 부호화 구조를 지원하는 비트 분배 방법을 제안한다. 그림 6은 제안 방법의 순서도를 나타낸다.

그림 6과 같이 먼저 IDR 주기에 대한 목표 비트를 계산한다. 주어진 목표 비트율은 1초에 사용되어야 할 비트량 이므로 IDR 주기에 대한 목표 비트 TB_{IDR} 는 아래와 같이 계산된다.

$$TB_{IDR} = \frac{TR_{seq}}{fps} \cdot N_{IDR} \quad (8)$$

여기서 TR_{seq} 는 전체 시퀀스에 대해 주어진 목표 비트 율이고 fps 는 주어진 시퀀스의 초당 프레임 율 (frame per second)이고 N_{IDR} 은 IDR 주기의 크기이다. 본 논문에서는 $N_{IDR} = 32$ 로 가정한다. IDR 프레임이 먼저 부호화 되고 나면 TB_{IDR} 로부터 IDR 프레임의 출력 비트를 제외한 나머지 비트를 4개의 GOP에 프레임 중요도를 고려하여 분배한다. 여기서 중요도란 표1에 나타난 ω 를 의미한다. 예를

들어, 그림 4와 같이 IDR 주기가 32이고 GOP 크기가 8인 경우에 GOP 1부터 GOP 4까지는 아래 식으로 계산되는 GOP별 목표 비트, TB_{GOP1} , TB_{GOP2} , TB_{GOP3} , TB_{GOP4} 를 할당 받는다.

$$TB_{GOP1} = (TB_{IDR} - B_{fr0}) \cdot \frac{\sum_{i=1}^8 \omega_i}{\sum_{i=1}^{31} \omega_i} \quad (9)$$

$$TB_{GOP2} = (TB_{IDR} - B_{fr0}) \cdot \frac{\sum_{i=9}^{16} \omega_i}{\sum_{i=1}^{31} \omega_i} \quad (10)$$

$$TB_{GOP3} = (TB_{IDR} - B_{fr0}) \cdot \frac{\sum_{i=17}^{24} \omega_i}{\sum_{i=1}^{31} \omega_i} \quad (11)$$

$$TB_{GOP4} = (TB_{IDR} - B_{fr0}) \cdot \frac{\sum_{i=25}^{31} \omega_i}{\sum_{i=1}^{31} \omega_i} \quad (12)$$

식 (9)-(12)에서 B_{fr0} 는 IDR 프레임이 부호화되어 실제로 출력된 출력 비트를 의미하고 분모의 $\sum_{i=1}^{31} \omega_i$ 와 분자의 ω 의 합은 각각 IDR 전체의 중요도와 해당 GOP의 중요도를 나타낸다. 여기서 중요한 점은 GOP마다 분배된 목표 비트는 한번 분배된 이상 상호간에 조정되지 않는다. 기존의 율 제어 방법들에서는 GOP 단위를 넘어서 출력 비트와 목표

비트의 차이가 전달되도록 비트 예산을 관리했지만, 병렬 부호화 구조에서는 여러 GOP가 동시에 부호화되므로 GOP 내에서 발생한 오차가 다른 GOP로 전달되지 않도록 하였다.

GOP 마다 목표 비트가 분배되었으면, 각 GOP에서는 GOP내의 각 프레임에 프레임 계층 별 중요도에 따라 비트를 할당한다. 예를 들어 GOP 1의 경우, 아래 과정으로 각 프레임에 비트를 할당한다. 먼저 아래 식으로 8번 프레임에 대한 목표 비트를 계산한다.

$$TB_8 = TB_{GOP1} \cdot \frac{\omega_8}{\sum_{i=1}^8 \omega_i} \quad (13)$$

8번 프레임을 부호화한 이후 실제로 출력된 비트인 B_{fr8} 에 대해 다음과 같이 4번 프레임에 대한 목표 비트를 계산한다.

$$TB_4 = (TB_{GOP1} - B_{fr8}) \cdot \frac{\omega_4}{\sum_{i=1}^7 \omega_i} \quad (14)$$

이후, 2번 프레임과 6번 프레임에 대한 목표 비트는 다음과 같이 계산된다.

$$TB_2 = TB_6 = (TB_{GOP1} - B_{fr8} - B_{fr4}) \cdot \frac{\omega_2}{\sum_{i=1}^3 \omega_i + \sum_{i=5}^7 \omega_i} \quad (15)$$

여기서 2번 프레임과 6번 프레임은 같은 프레임 계층이기 때문에 $\omega_2 = \omega_6$ 이다. 마지막으로 1, 3, 5, 7번 프레임에 할당되는 목표 비트는 아래와 같이 계산된다.

$$TB_1 = TB_3 = \left\{ \frac{(TB_{GOP1} - B_{fr8} - B_{fr4})}{2} - B_{fr2} \right\} \cdot \frac{\omega_1}{\omega_1 + \omega_3} \quad (16)$$

$$TB_5 = TB_7 = \left\{ \frac{(TB_{GOP1} - B_{fr8} - B_{fr4})}{2} - B_{fr6} \right\} \cdot \frac{\omega_5}{\omega_5 + \omega_7} \quad (17)$$

여기서도 (15)와 마찬가지로 1, 3, 5, 7번 프레임은 모두 같은 프레임 계층이므로 $\omega_1 = \omega_3 = \omega_5 = \omega_7$ 이다. 이와 같이 부호화 과정과, 남은 프레임에 대한 비트 분배 과정을 반복하며 한 GOP의 율 제어 과정을 종료한다. 한편 율 모델의 부정확성으로 인해 필연적으로 목표 비트와 출력 비트 간에는 오차가 발생한다. 계층 4를 제외한 나머지 계층에서 발생하는 오차는 (14)-(17)에 의해 더 깊은 계층으로 전달되어 그 차이를 해결할 수 있다. 그러나 계층 4에 속하는 프레임들로부터 발생하는 오차는 해결할 수 없게 되는 문제가 있다. 병렬 부호화가 아닌 순차적인 구조를 갖는 부호화기의 경우는 계층 4에 속하는 프레임으로부터 발생하는 오차가 그 다음 GOP의 계층 1에 속하는 프레임으로 전달될 수 있다. 예를 들어 그림 4와 같은 GOP 구조를 갖는 경우 7번 프레임에서 발생한 오차를 16번 프레임의 목표 비트에 반영하여 해결할 수 있다. 그러나 본 논문에서 가정하는 병렬 부호화기의 경우 여러 개의 GOP가 병렬적으로 동시에 부호화되기 때문에 계층 4에서 발생하는 오차를 그 다음 GOP에서 해결할 수가 없다. 예를 들어 그림 4와 같은 GOP 구조에서 7번 프레임에서 오차가 발생했다 해도, 이미 16번 프레임은 그와는 독립적인 목표 비트를 가지고 부호화가 시작되어있는 상태이다. GOP 간의 부호화 의존성이 (coding dependency) 계층 1의 프레임들에서만 존재하기 때문에 (8, 16, 24번 프레임) 목표비트와 출력비트의 오차가 GOP 단위를 넘어서 전달될 수가 없고, 결국 계층 4에 포함된 프레임에서 발생하는 오차는 해결될 수가 없다. 그러나 설명한 바와 같이 이는 병렬 부호화 구조의 필연적인 문제이다.

2. 모델 파라미터 업데이트

병렬 부호화 구조를 적용하면 비트 예산 관리에서 나타나는 문제 외에도, 모델 파라미터 업데이트에서도 문제가 발생한다. R-λ모델 율 제어 알고리즘은 같은 프레임 계층은 같은 모델 파라미터를 공유하도록 되어 있다. 순차적인 부호화 구조의 경우 각 프레임 계층의 초기 프레임들에서는 모델이 부정확하여 오차가 많이 발생하겠지만 부호화가 진행됨에 따라 모델 파라미터 업데이트를 통해 모델이 더

욱 정확해진다. 그러나 III장에서 설명한 병렬화 구조에서는 모델 파라미터가 IDR 주기 단위로 전달되지도 않을뿐더러, 예를 들어 프레임 계층이 4인 홀수 번째 프레임들의 경우 모두 독립적으로 부호화되기 때문에 어떤 프레임이 먼저 부호화된다는 순서상의 보장이 없고, 최악의 경우 모든 프레임이 모델 파라미터의 초기값 만으로 모델을 적용해야 하는 경우도 발생한다.

이 문제를 해결하기 위해 기존의 방식처럼 모델 파라미터를 순차적으로 업데이트 하지 않고 프레임 계층 마다 모델 파라미터를 저장해 두었다가, 부호화가 시작되면 해당 모델 파라미터를 이용하고, 부호화가 끝나면 부호화 결과에 맞게 모델 파라미터를 업데이트한다. 그 후에 저장되어 있는 모델 파라미터와 비교하여 업데이트 횟수가 더 많은 파라미터를 보존한다. 예를 들어 현재 계층 4에 저장되어 있는 모델 파라미터가 1, 3, 5, 7 번 프레임을 부호화 하면서 3번 업데이트 되어 있는 상태이고 이와 동시에 9, 11 번 프레임을 부호화 하면서 1번 업데이트 된 모델 파라미터가 있다면 3번 업데이트 되어 있는 모델 파라미터를 보존하도록 한다는 것이다. 이 같은 방법으로 업데이트가 많이 되어 현재 시퀀스에 적합한 모델 파라미터를 선택하도록 하였다.

3. 전처리 결과를 활용한 비트 분배

앞서 제안한 방법과 더불어 입력 영상에 대한 전처리 과정을 통해 영상의 복잡도 정보를 획득하고 이를 비트율 분배에 활용함으로써, 율 제어 성능을 더욱 향상시킬 수 있다. 본 절에서는 영상을 부호화하기 이전에 전처리 과정을 통하여 영상의 특성을 미리 파악하여 율 제어에 활용하는 방법을 제안한다. 영상이 부호화되기 이전에 한 IDR 주기 만큼을 미리 입력받아 16×16 블록 크기로 원본 영상 간의 움직임 추정을 수행한다. 이 때, 실제 영상이 부호화될 때 참조하는 프레임을 동일하게 사용하여, 각 블록의 SAD (sum of absolute difference)를 계산한다. 이렇게 계산된 SAD 값은 영상의 복잡도와도 연관이 있고, 부호화 효율과도 연관이 있을 것으로 판단하여 이 SAD 값을 프레임 전체에 대하여 평균을 취해 이 값으로 표 1의 ω 를 대체하였다. 이렇게 함으로써, 표 1과 같이 프레임의 계층에 따라 동일한 ω 값을 사용하는 것에 비해, 각 GOP의 영상 특성을 더욱

잘 반영할 수 있다.

한편, 영상의 해상도가 4K UHD (3840×2160)와 같이 매우 큰 입력 영상의 경우에는 실시간 부호화를 위해 HEVC의 슬라이스 구조를 활용하면 부호화 속도를 더욱 향상시킬 수 있다. 예를 들어 3840×2160 크기의 4K UHD 영상을 6개의 슬라이스로 분할하여 각 슬라이스를 독립적으로 병렬 부호화하면 부호화 시간이 크게 단축된다. 그러나 이렇게 슬라이스로 분할하여 독립적으로 부호화 하게 되면 슬라이스 간 화질 불균형이 발생할 수도 있고 심한 경우 슬라이스 경계가 시각적으로 관찰 될 수도 있다. 이러한 문제는 한 프레임을 6개의 슬라이스로 분할하면 영역에 따라 복잡한 영역을 포함하는 슬라이스와 단순한 영역을 포함하는 슬라이스로 서로 다른 특성을 갖는 슬라이스로 분할되더라도 불구하고 모두 동일한 비트를 할당받기 때문에 발생한다. 복잡한 영역을 포함하는 슬라이스는 상대적으로 낮은 화질로 부호화되고 단순한 영역을 포함하는 슬라이스는 상대적으로 높은 화질로 부호화되어 슬라이스 간 화질 불균형이 발생하고 심할 경우 슬라이스 경계가 시각적으로 관찰되게 된다. 이 문제를 해결하기 위해서도 마찬가지로 전처리 결과를 활용할 수 있다. 예를 들어 한 프레임을 6개의 슬라이스로 분할하였고 IDR 주기가 32라고 가정하면 한 IDR 주기 내에는 192(= 6×32)개의 슬라이스 영역이 존재한다. 전처리로부터 얻어진 SAD 값을 192 슬라이스에 대해 평균하여 그 값의 비율대로 비트를 최초 할당한다. 그 이후에는 IV 장 1절에서 설명한 비트 분배 방식으로 비트 율을 제어한다.

V. 실험결과

본 논문에서 제안하는 율 제어 방법의 성능을 검증하기 위하여 4K UHD (3840×2160) 실시간 부호화를 위한 병렬 구조의 소프트웨어 HEVC 인코더에 제안하는 율 제어 방법을 구현하여 실험을 수행하였다. 총 10개의 테스트 시퀀스에 대해 각 4개의 목표 비트 율을 할당하고 이에 대해 테스트를 수행 하였다. 표 2는 더 상세한 실험 조건을 나타낸다.

표 3에서는 표 2에서 제시한 실험 조건으로 실험한 결과 각 영상에 대한 목표 비트 율과 출력 비트 율을 제시하였다.

표 2. 실험 조건

Table 2. Experimental conditions

Sequence	10
Number of encoded frames	32
GOP size	8
Frame rate	60
Bit depth	10
YUV format	4:2:0
Max CU size	64×64
Profile	Main10

표 3에서 볼 수 있듯이 제안된 올 제어 방법에 의해 대부분의 영상에서 목표 비트율과 출력 비트율 간의 오차가 10%내외임을 볼 수 있다. 이를 통해 제안하는 비트 분배 방식이 효과적으로 작동한다는 것을 확인할 수 있었다. 그러나 오차가 20%이상 발생하는 테스트 영상들이 있었다. 이 같

이 큰 오차가 발생하는 이유로는 해당 테스트 시퀀스들의 길이가 1개의 IDR 주기인 32장에 불과하였고, 부호화가 동시에 시작되는 프레임이 많기 때문에 R-λ 모델의 모델 파라미터가 초기 값이 사용되는 경우가 많아 낮은 정확도의 R-λ 모델이 적용된 결과로 판단된다. 오차가 많이 발생하는 10번 영상의 경우 장면 전환을 포함하고 있어 더 많은 오차가 발생한 것으로 보여 진다. 다음으로는 전처리 결과를 활용하여 영상의 주관적 화질을 향상시키는 방법을 적용하였다. 영상을 슬라이스로 분할하여 IV.3절에서 설명한 대로 4K UHD 영상을 6개의 슬라이스로 분할하여, 각 슬라이스의 전처리 결과로 얻어진 SAD 값에 따라 슬라이스 별로 비트를 할당한다. 표 4는 전처리 결과를 활용하지 않고, 모든 슬라이스에 균일하게 비트를 할당했을 때와, 전처리

표 3. 제안한 비트율 제어 방법 실험결과

Table 3. Experimental result

Seq.	Target (kbps)	Output (kbps)	Error (%)	Target (kbps)	Output (kbps)	Error (%)	Target (kbps)	Output (kbps)	Error (%)	Target (kbps)	Output (kbps)	Error (%)
1	34183	29755	13	11779	7777	34	6629	5201	22	3942	3670	7
2	44924	53022	18	21553	24883	15	3942	3670	7	4256	4480	5
3	38810	34054	12	16229	16802	4	44924	53022	18	4533	4689	3
4	38119	38527	1	18104	19000	5	21553	24883	15	4660	4047	13
5	41088	49059	19	21749	26546	22	10071	11134	11	4670	5081	9
6	39927	38167	4	18283	16988	7	4256	4480	5	3939	3459	12
7	37277	36365	2	16960	17864	5	38810	34054	12	3670	4852	32
8	39086	27737	29	12684	8803	31	16229	16802	4	3960	3692	7
9	53409	57761	8	30951	32921	6	8165	7846	4	5374	7150	33
10	6261	6695	7	3383	3781	12	4533	4689	3	1005	2434	142

표 4. 전처리 결과 활용 여부에 따른 부호화 성능

Table 4. Performance of Encoding with respect to the use of preprocessing

Seq.	without preprocessing			with preprocessing		
	Target (kbps)	Output (kbps)	PSNR-Y (dB)	Target (kbps)	Output (kbps)	PSNR-Y (dB)
1	19,999.998	23,092.42	39.9070	19,999.997	18,454.8	39.2781
2		22,035.87	38.0683	19,999.996	19,081.01	38.2650
3		22,051.26	38.6307	19,999.997	19,278.99	39.1030
4		20,077.96	34.8815	19,999.998	19,624.38	34.6913
5		20,851.31	36.4836	19,999.998	21,196.38	37.1567
6		21,675.8	41.8547	19,999.996	17,835.84	41.3384
7		20,879.06	37.1371	19,999.997	19,455.69	36.3078
8		24,939.65	41.7070	19,999.998	17,146.08	43.2591
9		20,079.23	33.2931	19,999.997	20,873.22	32.2692
10		81,360.24	38.2269	19,999.996	28,370.42	41.7008

결과를 활용하여 비트를 서로 다르게 할당했을 때의 결과를 나타낸다. 표 4의 결과를 얻기 위해 실시간 지상파 방송을 가정하고 20Mbps의 목표 비트를 설정하였다. 전처리결과를 활용하지 않은 경우 슬라이스 6개에 모두 동일하게 3,333.333Mbps의 목표 비트를 할당하여 6개 슬라이스에는 도합 19,999.998kbps의 목표 비트가 할당되었다. 전처리결과를 활용한 경우에도 총 20Mbps의 목표 비트를 가정하였지만, 전처리 결과에 따라 6개의 슬라이스에 비트가 분배되

면서 정수 이하의 비트가 절삭되어 총 목표 비트량은 조금씩 차이가 발생하였다.

출력 비트율은 각 시퀀스당 6개 슬라이스 전체의 출력 비트율을 합하여 구한 값이고 PSNR-Y 값은 6개 슬라이스 각각의 PSNR 값을 평균하여 구한 값이다. 그림 7은 전처리결과를 활용하기 전과 후에 따라 사라지는 슬라이스 경계를 보인다.

그림 7에서 확인할 수 있듯이 슬라이스 경계가 전처리



그림 7. 전처리결과가 있을 때와 없을 때의 결과 영상
 Fig. 7. Result image without/with preprocessing

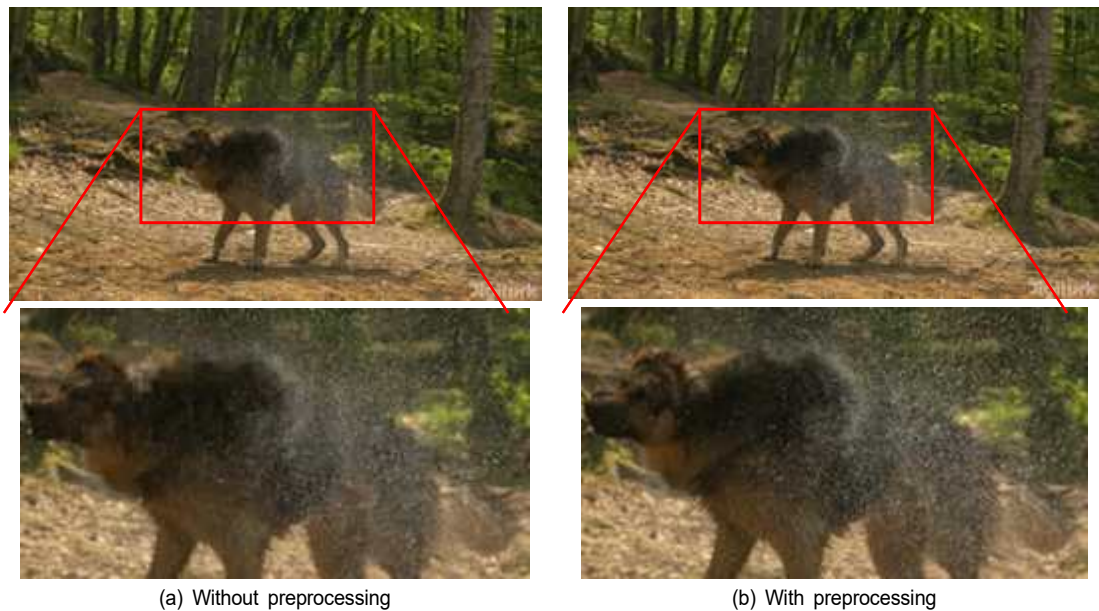


그림 8. 전처리결과가 있을 때와 없을 때의 결과 영상
 Fig. 8. Result image without/with preprocessing

결과를 활용하는 경우 사라지는 것을 볼 수 있다. 그림 8은 전처리 결과를 활용해 슬라이스 간 화질 균형을 이룬 결과를 보인다.

그림 8에 나타난 영상은 개가 물을 터는 장면인데, 배경 부분은 거의 움직임이 없고 중앙에 있는 개의 몸통 털 부분에서 물이 사방으로 튀면서 매우 복잡한 움직임을 나타낸다. 이 경우에 모든 슬라이스에 동일한 비트가 할당되면 단순한 텍스처를 갖는 슬라이스에 필요이상의 많은 비트가 할당되고 복잡한 텍스처를 갖는 슬라이스에는 필요이하의 비트가 할당된다. 그러나 전처리 결과를 활용하여 복잡도가 높은 슬라이스에 더 많은 비트를 할당하도록 하면 그림 8에서 볼 수 있듯이 전반적으로 화질 밸런스가 맞으면서 화질이 향상되는 것을 확인할 수 있었다.

VI. 결 론

본 논문에서는 프레임 수준, GOP 수준 및 IDR 주기 수준의 병렬화 구조가 적용된 HEVC 부호화기에 적용할 수 있는 율 제어 방법을 제안하였다. 제안하는 율 제어 알고리즘은 기존의 R- λ 모델을 기반으로 하되 병렬화에 적합하도록 새로운 율 분배 방식을 개발하였다. 기존의 R- λ 모델 율 제어 방법을 포함한 대부분의 율 제어 방법에서 사용하던 율 분배 방식은 순차적 부호화를 가정하고 설계되었기 때문에 병렬화 구조에 적용하는 데에는 제약 사항이 존재하였다. 이를 극복하기 위해 본 논문에서는 GOP 별로 목표 비트를 할당하고, 오차 전달 방향을 순차적인 부호화 순서가 아닌 계층적 B 구조의 프레임의 계층 방향으로 전달하고, GOP 간에는 비트 발생 오차를 전달하지 않도록 함으로써 해결할 수 있었다. 나아가 전처리를 통해 영상의 복잡도를 계산하고 이를 사전 정보로 활용하여 율 제어 정확도는 물론 주관적 화질을 향상시키는 방법을 제안하였다. 실험 결과

를 통해 본 제안 방법의 타당성을 검증하였다.

참 고 문 헌 (References)

- [1] B. Bross, W.-J. Han, G. J. Sullivan, J.-R. Ohm, and T. Wiegand, *High Efficiency Video Coding (HEVC) Text Specification Draft 9, document JCTVC-K1003*, ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC), Oct. 2012.
- [2] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649 - 1668, Dec. 2012.
- [3] ITU-T, *Advanced Video Coding for Generic Audio-Visual Services, ITU-T Rec. H.264 and ISO/IEC 14496-10 (AVC)*, ITU-T and ISO/IEC JTC 1, May 2003 (and subsequent editions).
- [4] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560 - 576, Jul. 2003.
- [5] J. H. Jang, and C. E. Rhee, "Down Sampling for Fast Rough Mode Decision for a Hardware-based HEVC Intra-frame encoder," *Journal of Broadcast Engineering*, vol. 21, no. 3, pp.341-348, May 2016.
- [6] S. Jeon, N. Kim, and B. Jeon, "CU Depth Decision Based on FAST Corner Detection for HEVC Intra Prediction," *Journal of Broadcast Engineering*, vol. 21, no. 4, pp. 484-492, July 2016.
- [7] J. Lim, Y.-J. Ahn, and D. Sim, "Study on Fast HEVC Encoding with Hierarchical Motion Vector Clustering," *Journal of Broadcast Engineering*, vol. 21, no. 4, pp.578-591, July 2016.
- [8] Y. Kim, J. Seok, S.-h. Jung, H. Kim, and J. S. Choi "Tile-level and Frame-level Parallel Encoding for HEVC," *Journal of Broadcast Engineering*, vol. 20, no. 3, pp. 388-397, May 2015.
- [9] B. Li, H.Li, L. Li and J. Zhang, *Rate control by R-lambda model for HEVC, document JCT-VC K0103*, Shanghai, CN, Oct. 2012.
- [10] B. Li, H.Li, L. Li and J. Zhang, " λ Domain Rate Control Algorithm for High Efficiency Video Coding," *IEEE Trans. Image Processing*, Vol. 23, No. 9, Sept. 2014.
- [11] H. Choi, J. Nam, J. Yoo, D. Sim, and I. Baji'c, *Rate Control Based on Unified RQ Model for HEVC, document Rec. JCTVC-H0213*, San Jose, CA, USA, Feb. 2012.
- [12] J. Si, S. Ma, W. Gao, and M. Yang, *Adaptive Rate Control for HEVC, document Rec. JCTVC-I0433*, Geneva, Switzerland, Apr./May 2012.
- [13] (2017, Jan. 12). HM, HEVC Test Model [Online]. Available: http://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/.

저 자 소 개



김 대 은

- 2011년 2월 : 고려대학교 전기전자전파공학부 학사
- 2014년 2월 : 한국과학기술원 전기및전자공학과 석사
- 2014년 3월 ~ 현재 : 한국과학기술원 전기및전자공학과 박사과정
- ORCID : [hhttp://orcid.org/0000-0003-0948-7049](http://orcid.org/0000-0003-0948-7049)
- 주관심분야 : HDR 영상 처리, 비디오 부호화



장 용 준

- 1997년 8월 : 서울대학교 산업공학과 공학사
- 2000년 8월 : 위스콘신 주립대학교 산업공학과 석사
- 2012년 8월 : 한국과학기술원 정보통신공학과 박사
- 2012년 4월 ~ 2015년 6월 : 서울아산병원/울산대학교 의학과 리서치펠로우
- 2015년 7월 ~ 2016년 7월 : 한국과학기술원 연구조교수
- 2016년 8월 ~ 현재 : 스웨덴 왕립공과대학교 방문연구원
- ORCID : [hhttp://orcid.org/0000-0001-8953-3986](http://orcid.org/0000-0001-8953-3986)
- 주관심분야 : 비디오 코딩, 머신 러닝, 의료영상처리



김 문 철

- 1989년 2월 : 경북대학교 전자공학과 학사
- 1992년 12월 : University of Florida, Dept. of Electrical and Computer Engineering, 석사
- 1996년 8월 : University of Florida, Dept. of Electrical and Computer Engineering, 박사
- 1997년 1월 ~ 2001년 1월 : 한국전자통신연구원 방송비디어 연구부, 선임연구원
- 2001년 2월 ~ 2009년 2월 : 한국정보통신대학교 공학부 조교수/부교수
- 2009년 3월 ~ 현재 : 한국과학기술원 전기및전자공학과 부교수/정교수
- ORCID : [hhttp://orcid.org/0000-0003-0146-5419](http://orcid.org/0000-0003-0146-5419)
- 주관심분야 : Perceptual Video Coding, SDR/HDR Image/Video Quality Assessment and Modeling, Super-Resolution, Image/Video Analysis and Understanding, Pattern Recognition, Machine Learning



임 웅

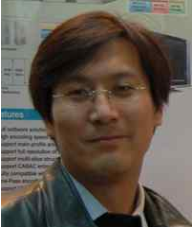
- 2008년 2월 : 광운대학교 컴퓨터공학과 학사
- 2010년 2월 : 광운대학교 컴퓨터공학과 석사
- 2016년 2월 : 광운대학교 컴퓨터공학과 박사
- 2016년 4월 ~ 현재 : 한국전자통신연구원 실감AV연구그룹 연구원
- ORCID : [hhttp://orcid.org/0000-0002-1772-0683](http://orcid.org/0000-0002-1772-0683)
- 주관심분야 : 디지털 영상처리, 디지털 비디오 압축



김 휘 용

- 1994년 8월 : 한국과학기술원(KAIST) 전기및전자공학과 (공학사)
- 1998년 2월 : 한국과학기술원(KAIST) 전기및전자공학과 (공학석사)
- 2004년 2월 : 한국과학기술원(KAIST) 전기및전자공학과 (공학박사)
- 2003년 8월 ~ 2005년 10월 : (주)에드팩테크놀로지 기술연구소 멀티미디어팀장
- 2006년 9월 ~ 2010년 8월 : 과학기술연합대학원대학교(UST) 이동통신및디지털방송공학과 겸임교수
- 2013년 9월 ~ 2014년 8월 : Univ. of Southern California(USC) 멀티미디어통신연구실 방문연구원
- 2005년 11월 ~ 현재 : 한국전자통신연구원(ETRI) 미디어연구본부 실감AV연구그룹장
- ORCID : [hhttp://orcid.org/0000-0001-7308-133X](http://orcid.org/0000-0001-7308-133X)
- 주관심분야 : 영상 압축, 컴퓨터 비전, 멀티미디어 시스템, 실감미디어서비스

저 자 소 개



석진욱

- 1993년 : 홍익대학교 전기제어공학과 학사
- 1995년 : 홍익대학교 전기제어공학과 석사
- 1998년 : 홍익대학교 전기공학과 박사
- 2000년 ~ 현재 : 한국전자통신연구원 실감AV연구그룹 책임연구원
- ORCID : <http://orcid.org/0000-0001-5318-1237>
- 주관심분야 : 비선형 확률시스템, 영상 압축, UHDTV 방송 시스템