

일반논문 (Regular Paper)

방송공학회논문지 제24권 제3호, 2019년 5월 (JBE Vol. 24, No. 3, May 2019)

<https://doi.org/10.5909/JBE.2019.24.3.463>

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

양방향 예측 모드를 위한 저복잡도 LIC 방법 연구

최한솔^{a)}, 변주형^{a)}, 방건^{b)}, 심동규^{a)*}

Low-complexity Local Illuminance Compensation for Bi-prediction mode

Han Sol Choi^{a)}, Joo Hyung Byeon^{a)}, Gun Bang^{b)}, and Dong Gyu Sim^{a)*}

요약

본 논문에서는 양방향 화면 간 예측에서 LIC(Local Illuminance Compensation)의 복잡도 감소를 위한 방법을 제안한다. LIC는 화면 간 예측의 정확도를 높이기 위해 현재 블록과 참조블록의 주변 복원샘플을 이용하여 지역 조명 보상을 수행한다. 지역 조명 보상을 위해 필요한 가중치와 오프셋을 주변 복원 샘플을 이용하여 부/복호화기 양측에서 계산하기 때문에 별도의 정보 전송 없이 부호화 효율이 향상되는 장점이 있지만 부호화 예측 단계 및 복호화 단계에서 가중치와 오프셋을 구하기 때문에 부/복호화 복잡도가 높아지는 단점을 가지게 된다. 따라서 본 논문에서는 저 복잡도 LIC를 위해 크게 두 가지 방법을 제안한다. 첫 번째 방법은 양방향 예측 시 가중치 없이 오프셋만으로 조명보상을 적용하는 방법이고, 두 번째는 양방향 예측에서 LIC를 양방향예측을 통해 구해진 참조 블록의 가중 평균 단계 이후 적용하는 방법이다. 제안하는 방법의 성능 평가를 위하여 RA(Random Access) 조건에서 MPEG 표준 실험 영상의 B, C, D 클래스를 이용하여 BMS-2.0.1과 BD-rate 성능을 비교한다. 실험결과로서 본 논문에서 제안하는 방법이 BMS-2.0.1 대비 BD-rate 성능 관점에서 Y, U, V 각각에 대하여 평균 0.29%, 0.23%, 0.04%의 BD-rate 결과를 보이고 부/복호화 시간은 거의 동일하다. BD-rate의 손실이 있었지만, LIC 파라미터 유도과정에서 곱셈 연산이 제거되고 덧셈 연산이 절반으로 감소됨에 따라 LIC의 계산 복잡도가 크게 감소되었다.

Abstract

This paper proposes a method for reducing the complexity of LIC (Local Illuminance Compensation) for bi-directional inter prediction. The LIC performs local illumination compensation using neighboring reconstruction samples of the current block and the reference block to improve the accuracy of the inter prediction. Since the weight and offset required for local illumination compensation are calculated at both sides of the encoder and decoder using the reconstructed samples, there is an advantage that the coding efficiency is improved without signaling any information. Since the weight and the offset are obtained in the encoding prediction step and the decoding step, encoder and decoder complexity are increased. This paper proposes two methods for low complexity LIC. The first method is a method of applying illumination compensation with offset only in bi-directional prediction, and the second is a method of applying LIC after weighted average step of reference block obtained by bidirectional prediction. To evaluate the performance of the proposed method, BD-rate is compared with BMS-2.0.1 using B, C, and D classes of MPEG standard experimental image under RA (Random Access) condition. Experimental results show that the proposed method reduces the average of 0.29%, 0.23%, 0.04% for Y, U, and V in terms of BD-rate performance compared to BMS-2.0.1 and encoding/decoding time is almost same. Although the BD-rate was lost, the calculation complexity of the LIC was greatly reduced as the multiplication operation was removed and the addition operation was halved in the LIC parameter derivation process.

Keyword : VVC, Local Illumination Compensation, Complexity, Simplification

Copyright © 2016 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

개인 멀티미디어 기기 및 각종 고성능 디스플레이 장치의 보편화와 통신 서비스 기술의 발전에 따라 4K 이상의 초고해상도 멀티미디어 콘텐츠의 서비스가 급격히 성장하고 있다. 점차 고화질/고해상도 영상에 대한 수요가 증가함에 따라 이를 한정된 저장장소에 저장하고 원활하게 전송하기 위한 영상 압축 기술이 요구되고 있다. ISO/IEC MPEG(Moving Picture Expert Group)과 ITU-T VCEG(Video Coding Expert Group)은 이전 비디오 압축 표준인 HEVC(High Efficiency Video Coding) 대비 2배 이상의 압축 효율 향상을 목표로 JVET(Joint Video Exploration Team)을 구성하여 VVC(Versatile Video Coding)^[1]라고 불리는 차세대 비디오 압축 표준화를 진행하고 있다. 차세대 비디오 압축 표준의 레퍼런스 소프트웨어인 VTM(VVC Test Model)^[2]은 이전 표준의 블록 기반의 하이브리드 구조를 바탕으로 압축 효율 향상을 위해 검증된 최소한의 기술들을 포함하고 있다. VTM에 포함되어 있지는 않지만 압축 효율 향상에 도움이 될 후보 기술들이 BMS(Benchmark Set)^[3]에 포함되어 개발되고 있다. 차세대 비디오 압축 표준은 압축 효율 향상에 방해가 되는 프레임 간 조명 변화를 해결하고자 노력 중이다. 조명 변화에 의한 부호화 효율 감소의 문제는 다시점 비디오 압축 시 시점이 다른 영상으로부터의 예측에서 빈번히 발생한다^[4]. 또한 동일한 시점의 영상에서도 자연광, 인공조명, 그림자 등의 빈번한 변화에

의하여 압축 효율이 감소한다. 화면 간 예측을 수행할 때 레퍼런스 프레임에서 현재 블록과 픽셀 값의 차이가 가장 작은 블록을 찾는다. 이때 현재 프레임과 참조 프레임 간의 지역적 또는 전역적인 조명 변화가 있을 경우 변화된 조명 값에 영향을 받아 정확한 모션을 찾을 수 없고 조명 변화가 없는 경우보다 예측 에러가 커질 수 있다. 이전 표준인 H.264/AVC^[5]와 HEVC^[6]에는 이러한 문제를 해결하기 위해 WP(Weighted Prediction)^[7] 기술이 존재하였다. WP는 인터 슬라이스의 예측 신호에 전역 조명 보상을 통해 원본 신호와 예측 신호의 오차를 줄이는 기술이다. 슬라이스 당 보상 파라미터를 지정하여 전송한다. 현재 프레임과 참조 프레임 간의 밝기 변화가 프레임 전체에 균일하다면 슬라이스 내 코딩 블록들을 동일한 파라미터로 밝기를 보상하여도 효과가 있다. 하지만 현재 프레임과 참조 프레임 간에 지역적인 밝기차가 있을 경우 하나의 파라미터를 사용하여 매크로블록 각각의 밝기차를 보상하기 어렵다^[8]. 이러한 WP 기술의 단점을 보완하여 차세대 비디오 압축 초기 레퍼런스 소프트웨어 JEM(Joint Exploration Model)^[9]에 LIC^[10]이 도입되었다.

JEM LIC는 화면 간 예측 기술 중 하나이며 예측 블록에 밝기를 보상해주어 프레임의 지역적인 밝기 차를 보상하는 것이 목적이다. LIC는 CU(Coding Unit)단위로 선택적으로 수행할 수 있으며 LIC를 수행할 시 단방향 예측일 경우와 양방향 예측일 경우에 예측 블록이 다음과 같은 방법으로 생성된다. 단방향 예측일 경우, 첫 번째 레퍼런스 프레임 리스트의 프레임으로부터 초기 예측 블록을 생성한다. 부호화기에서는 초기 예측 블록을 생성할 시 조명 조건을 고려하지 않기 위해 참조 블록과 분산의 차가 가장 작은 블록을 탐색한다. 초기 예측 블록에 보간을 수행하여 서브 픽셀 단위의 정교한 예측 블록을 생성한다. 그 다음 예측 블록에 LIC를 수행하여 최종 예측 블록을 유도한다. 밝기를 보상하기 위한 파라미터는 현재 블록과 예측 블록의 이웃하는 픽셀들 사이의 선형 모델을 유도함으로써 얻어진다. 양방향 예측일 경우, 두 레퍼런스 프레임 리스트로부터 각각 초기 예측 블록을 유도한다. 각 예측블록에 단방향 예측일 경우와 같은 방법으로 움직임 보상과 LIC를 수행한 뒤 두 예측블록을 병합한다.

LIC는 큰 압축률 이득을 얻지만 높은 계산 복잡도 문제가 해결되지 않았다. 특히 양방향 예측일 경우 단방향 예측

a) 광운대학교 컴퓨터공학과(Dept. of Computer Engineering, Kwangwoon University)

b) 한국전자통신연구원(Electronics and Telecommunications Research Institute)

‡ Corresponding Author : 심동규(Dong Gyu Sim)

E-mail: dgsim@kw.ac.kr

Tel: +82-2-940-5470

ORCID: <http://orcid.org/0000-0002-2794-9932>

※ 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학 지원사업(2017-0-00096) 및 IITP ETRI지원사업인 “이머시브 미디어 전문연구실” 사업(출연처과제번호2018-0-00207)에 의해 연구되었음.

※ This research was supported by the MIST(Ministry of Science and ICT), under the National Program for Excellence in SW(2017-0-00096), supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation) and Immersive Media Research Laboratory(IITP-2018-0-00207) supervised by the IITP(Institute for Information & communications Technology Promotion).

· Manuscript received November 30, 2018; Revised February 25, 2019; Accepted February 25, 2019.

일 경우 대비 2배의 LIC 복잡도를 갖는다. 양방향 예측일 경우 단방향 예측일 경우 보다 예측 성능이 높아서 LIC를 사용하여 얻는 이익이 상대적으로 적지만 복잡도가 늘어나는 문제점이 있다. 따라서 본 논문에서는 양방향 예측에서 LIC의 복잡도를 줄이기 위해 크게 두 가지 방법을 제안한다. 첫 번째 방법은 양방향 예측에서 LIC를 양방향 예측을 통해 구해진 참조 블록의 가중 평균 단계 이후 적용하는 방법이다. 양방향 예측일 경우 두 예측블록에 각각 LIC를 적용하므로 단방향 예측일 경우 대비 2배의 LIC 복잡도를 갖는다. 따라서 양방향 예측에서 복잡도를 줄이기 위하여 움직임이 보상된 두 예측 블록을 병합한 뒤 LIC를 수행할 것을 제안한다. 제안하는 두 번째 방법은 양방향 예측 시 가중치는 사용하지 않고 오프셋만 사용하여 조명 보상을 적용하는 방법이다. 양방향 예측 시 두 레퍼런스 리스트로부터 시간적으로 전과 후에 있는 프레임을 모두 참조하여 예측블록을 생성한다. 따라서 첫 번째로 제안하는 방법과 같이 LIC 이전에 두 예측블록을 먼저 병합하면, 생성된 예측블록의 밝기가 원본블록에 근접할 수 있고, 이후 LIC를 수행하여 오프셋 값으로 남은 밝기 차를 보상할 수 있다. 제안하는 방법은 BMS-2.0.1 대비 Y, U, V 각각에 대하여 평균 0.29%, 0.23%, 0.04%의 BD-rate 성능을 보이고 부/복호화 시간은 거의 동일하다. 또한 LIC 파라미터를 유도하는 과정에서 곱셈 연산을 사용하지 않아도 되고 곱셈 연산의 계산 복잡도가 기존의 방법 대비 절반으로 줄었다.

본 논문의 구성은 다음과 같다. 2 장에서는 JEM에 존재 하였던 기존의 LIC의 기술 소개와 등장 배경을 설명하고, 3 장에서는 제안하는 LIC를 설명한다. 4 장에서는 실험 결과를 제시하고 5 장에서는 본 논문의 결론을 맺는다.

II. JEM LIC 기술 개요

프레임 간의 조명 차이로 인해 압축 효율이 낮아지는 것을 해결하기 위하여 H.264/AVC에 WP를 도입하였다^[7]. WP는 현재 프레임과 참조 프레임 간의 전역적인 밝기를 보상하는 기술이다. 움직임 예측 수행을 통해 생성한 초기 예측 블록에 움직임 보상을 수행한 뒤 WP를 수행하여 참조 블록과의 밝기 차를 보상한다. WP에는 두 가지 방식이 존

재하는데 프레임간 거리를 기준으로 파라미터를 유도하여 사용하는 implicit 모드와 슬라이스마다 밝기 보상 파라미터를 전송하는 explicit 모드이다. WP는 선택적으로 사용할 수 있고 implicit모드와 explicit 모드 중 어떤 모드를 사용하는지 PPS(Picture Parameter Set)에서 전송된다. implicit 모드는 B 슬라이스에서 사용이 가능하다. 두 참조 프레임 리스트로부터 생성한 예측 블록들을 가중 합산하는 과정에서 각 예측 블록에 적용할 가중치를 현재 프레임과 참조 프레임 간의 상대적인 POC(Picture Order Count) 차이 값에 반비례하도록 둔다. 시간적 거리가 가까운 참조 프레임과의 조명 차이가 더 적다는 가정으로 하여 가까운 참조 프레임에서 생성한 예측 블록에 더 큰 가중치를 적용한다. 부/복호화기에서 가중치를 동일한 방법으로 유도하여 사용할 수 있어서 별도로 전송하지 않는다. explicit 모드는 P와 B슬라이스에서 모두 사용이 가능하다. 슬라이스마다 밝기 보상 파라미터를 지정하여 슬라이스 헤더에서 전송한다. implicit 모드가 weighting factor 만으로 밝기 보상을 수행했던 것에 반면 explicit 모드는 offset 값도 사용한다. 파라미터를 전송하는 비용이 들지만 implicit 모드보다 정밀하게 밝기 보상을 할 수 있다. WP는 필요성에 따라서 지속적으로 연구가 되어 HEVC에도 채택되었다. HEVC^[6]에서도 H.264의 WP와 같이 슬라이스 내 모든 코딩 블록에 동일한 밝기를 보상하기 때문에 지역적인 밝기 변화는 보상하기 어렵다. 따라서 이 점을 보완하여 지역적 밝기 보상을 위한 기술로 LIC가 소개되었고 차세대 압축 표준의 후보 기술로 연구가 되고 있는 중이다.

LIC^[10]는 지역적 밝기를 보상하고자 CU 단위로 최적화된 밝기 보상을 수행한다. LIC를 적용할 때에는 움직임 예측 수행 시 밝기차를 제외하였을 때 현재 블록과의 오차가 가장 적은 예측 블록을 탐색하여 정확한 움직임을 찾는다. LIC를 수행하지 않을 때는 움직임 예측의 오차 측정 수단으로 단순히 두 블록의 픽셀의 차이 값을 더하는 SAD(Sum Absolute Difference)를 사용한다. LIC를 수행할 때는 현재 블록과 예측 블록의 DC값 차이를 제외하고 움직임 예측을 수행하기 위해 MRSAD(Mean Removed Sum Absolute Difference)를 사용한다. 이와 같이 LIC를 수행할 경우와 수행하지 않는 경우 움직임 예측을 각각 수행해야 하므로 부호화기의 복잡도가 증가한다. 단방향 예측과 양방향 예측 모드 모두에서 선택적으로 적용될 수 있고 수행 여부가 1 비트

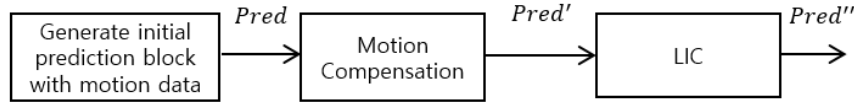


그림 1. 복호화기에서 단방향 예측이고 LIC를 수행할 때 움직임 예측 블록 생성의 흐름도
Fig. 1. Flow of LIC in uni-prediction in decoder

를 사용하여 CU마다 전송된다.

단방향 예측일 경우 예측 블록 생성 과정에서 LIC가 그림 1과 같은 순서로 수행된다. 부호화기에서는 움직임 예측을 수행하여, 복호화기에서는 부호화 시 정해진 움직임 벡터를 사용하여 초기 예측 블록을 생성한다. 정밀한 움직임 예측을 위해 정수 픽셀 단위의 예측 블록을 소수 픽셀 단위로 움직임 보상을 수행한다. LIC를 수행하는 경우에는 LIC와 효과가 일부 중복되는 WP를 수행하지 않고 LIC를 수행한다.

현재 블록과 예측 블록의 샘플 값들이 선형의 관계라는 가정으로 움직임 보상이 적용된 예측 블록에 선형 모델을 사용하여 식 (1)과 같이 LIC를 적용한다. α 는 scale factor이고, β 는 offset이다.

$$Pred'' = \alpha \cdot Pred' + \beta \quad (1)$$

CU마다 최적화된 밝기보상 파라미터를 유도하여 적용한

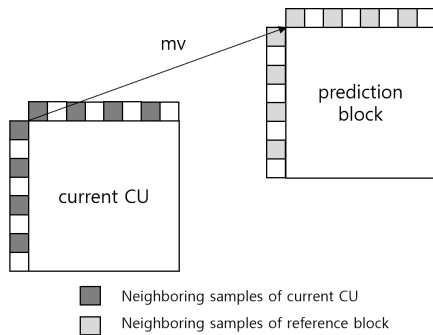


그림 2. 단방향 예측에서 LIC 파라미터 유도에 사용되는 픽셀
Fig. 2. Pixels used to derive LIC parameters in uni-directional prediction

다. $C(n)$ 은 현재 블록의 이웃하는 샘플들이고 $R(n)$ 은 참조 블록의 이웃 샘플들이다. 현재 블록을 부/복호화 하고자 하는 시점에 이미 부/복호화가 완료된 위쪽, 왼쪽으로 이웃하는 샘플을 이용하고 그림 2와 같이 절반의 갯수로 샘플링하여 사용한다.

$$\alpha = \frac{N \cdot \sum (C(n) \cdot R(n)) - \sum C(n) \cdot \sum R(n)}{N \cdot \sum (C(n) \cdot C(n)) - \sum C(n) \cdot \sum C(n)} \quad (2)$$

$$\beta = \frac{\sum R(n) - \alpha \cdot \sum C(n)}{N}$$

양방향 예측일 경우 예측 블록 생성 과정에서 LIC가 그림 3과 같은 순서로 수행된다. 참조 프레임 리스트들로부터 초기 예측 블록들(Pred0과 Pred1)을 유도하고 움직임 보상을 적용한 뒤 LIC를 수행한다.

두 예측 블록에 대하여 각각의 scaling factor α , α' 와 offset β , β' 를 유도하여 식 (3)과 같이 각 예측 블록에 LIC를 수행한다. 그림 4와 같이 첫 번째 예측 블록의 파라미터를 유도할 때는 현재 블록과 첫 번째 예측 블록의 이웃하는 샘플을 사용하고 두 번째 블록의 파라미터를 유도할 때는 두 번째 블록과 현재 블록의 이웃하는 샘플을 사용한다. 단방향 예측일 때와 동일하게 최소자승법을 사용하여 scaling factor와 offset을 유도한다. LIC가 수행된 경우에는 WP 단계에서 두 예측 블록의 가중치를 동일하게 하여 최종 예측 블록을 생성한다.

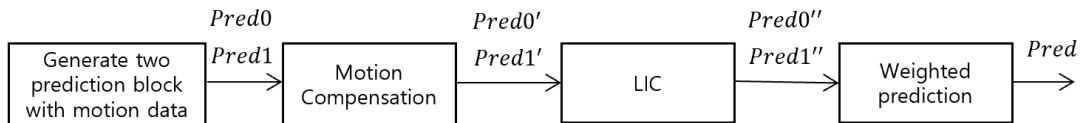


그림 3. 복호화기에서 양방향 예측이고 LIC를 수행할 때 흐름도
Fig. 3. Flow of LIC in bi-prediction in decoder

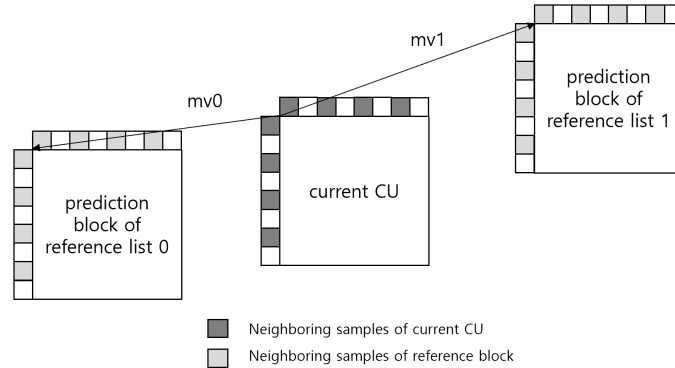


그림 4. 양방향 예측에서 LIC 파라미터 유도에 사용되는 픽셀
Fig. 4. Pixels used to derive LIC parameters in bi-directional prediction

$$\begin{aligned} Pred0'' &= \alpha * Pred0' + \beta \\ Pred1'' &= \alpha' * Pred1' + \beta' \end{aligned} \quad (3)$$

III. 제안하는 양방향 예측 모드에 대한 저복잡도 LIC

LIC는 큰 BD-rate 성능 향상을 거두었지만 부/복호화 시간 또한 크게 증가시켰다. 특히 단방향 예측일 경우 하나의 레퍼런스 프레임 리스트를 참조하여 생성한 예측블록에 LIC를 적용하는 반면 양방향 예측일 경우 두 레퍼런스 프레임 리스트로부터 각각 생성한 예측 블록에 LIC를 적용하기 때문에 LIC의 복잡도가 2배로 높다. 따라서 본 논문에서는 양방향 예측의 LIC 복잡도를 줄이고자 두 가지 방법을 제안한다. 첫 번째로 양방향 예측 시 그림 5와 같은 순서로 LIC를 수행하는 방법을 제안한다. 화면 간 예측 수행 시 현재 프레임보다 먼저 부/복호화 되어 DPB(Decoded Picture Buffer)에 저장되어있는 프레임들을 참조하여 예측 블록을 생성한다. 이때 DPB에 저장되어있는 프레임들로 레퍼런스 프레임 리스트를 2개를 구성한 뒤 각각의 레퍼런스 프레임 리스트로부터 두 개의 초기 예측 블록을 생성한다. 초기 예측블록들에 대하여 각각 보간을 수행하여 소수 픽셀 단위

의 정밀한 예측 블록을 생성하고 이를 움직임 보상이라 한다. 기존의 방법은 움직임 보상을 수행한 뒤 각 예측 블록에 LIC를 적용하였지만 제안하는 방법은 두 예측 블록을 병합하는 WP를 먼저 수행한다. 식 (4)와 같이 WP 단계에서 두 예측 블록이 동일한 비율로 가중 합산되어 병합된다.

$$Pred = (Pred0' + Pred1' + 1) \gg 1 \quad (4)$$

병합된 하나의 예측 블록에 식 (5)와 같이 LIC를 수행하여 최종 예측 블록을 생성한다.

제안하는 두 번째 방법은 LIC 파라미터 유도 과정의 계산 복잡도를 줄이기 위하여 식 (5)와 같이 scaling factor는 1로 고정하고 offset(β) 값만을 사용하여 간소화된 방법으로 밝기를 보상하는 방법이고 첫 번째 방법과 함께 사용된다. 양방향 예측일 경우 단방향 예측에 비해 더 많은 정보를 이용하여 오차가 더 적은 예측블록을 생성하므로 간소화된 LIC를 통해서도 충분히 밝기 보상을 수행할 수 있다.

$$Pred' = Pred + \beta \quad (5)$$

Offset 파라미터는 최소자승법을 사용하여 식 (6)을 통해

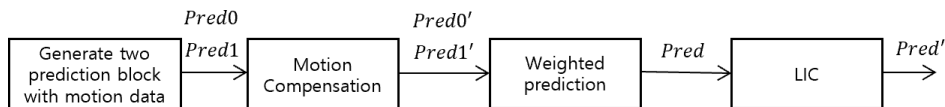


그림 5. 복호화기에서 제안하는 움직임 예측 블록 생성의 흐름도
Fig. 5. Proposed flow of LIC in bi-prediction in decoder

유도된다. WP에서 두 예측 블록을 평균내어 LIC를 적용하므로 리스트 0과 1로부터 생성된 두 예측 블록의 이웃하는 샘플들(R_0 , R_1)의 평균값과 현재 블록의 이웃하는 샘플들(C)의 값을 이용하여 offset을 유도한다. 파라미터 유도에 사용되는 샘플들의 위치는 기존의 LIC와 동일하다.

$$\beta = \frac{\sum((R_0(n) + R_1(n) + 1) \gg 1) - \sum C(n)}{N} \quad (6)$$

IV. 실험 결과

본 논문에서 제안하는 방법은 BMS-2.0.1을 기반으로 구현되었다. 성능 평가를 위한 실험은 MPEG 표준 실험 영상의 B, C, D 클래스를 사용하여 JVET CTC(Common Test Condition)^[11]의 RA(Random Access) 조건에서 실험을 진행하였다. 양자화 파라미터(QP)는 22, 27, 32, 37을 사용하였다. 본 논문에서는 속도 성능 관점에서 공정한 비교를 위

표 1. BMS2.0.1(with LIC on) 대비 제안하는 방법 1과 2를 모두 사용한 경우의 BD-Rate 성능 및 부/복호화 시간
Table 1. BD-Rate performance and coding time of proposed method 1 and 2 compared to BMS2.0.1(with LIC on)

Class	Sequence	BD-rate			Enc Time(%)	Dec Time(%)
		Y (%)	U (%)	V (%)		
Class B	MarketPlace	0.37%	0.35%	0.28%	101	99
	RitualDance	0.61%	0.18%	0.18%	101	99
	Cactus	0.42%	0.19%	0.02%	101	99
	BasketballDrive	0.35%	0.32%	0.36%	101	98
	BQTerrace	0.23%	-0.24%	-1.48%	101	103
Class C	BasketballDrill	0.42%	0.43%	0.39%	101	100
	BQMall	0.19%	0.20%	-0.07%	101	99
	PartyScene	0.28%	0.21%	0.21%	102	98
	RaceHorses	0.34%	0.34%	0.59%	101	102
Class D	BasketballPass	0.12%	0.02%	0.11%	102	97
	BQSquare	0.01%	0.52%	-0.42%	103	100
	BlowingBubbles	0.31%	0.14%	0.27%	101	100
	RaceHorses	0.08%	0.28%	0.11%	101	98
Overall	All	0.29	0.23	0.04	101	99

표 2. BMS2.0.1(with LIC on) 대비 제안하는 방법 1을 사용한 경우의 BD-Rate 성능 및 부호화 시간
Table 2. BD-Rate performance and coding time when using proposed method 1 compared to BMS2.0.1(with LIC on)

Class	Sequence	BD-rate			Enc Time(%)	Dec Time(%)
		Y (%)	U (%)	V (%)		
Class B	MarketPlace	0.41	0.46	0.27	101	99
	RitualDance	0.60	0.35	0.51	101	99
	Cactus	0.44	0.09	0.11	101	99
	BasketballDrive	0.34	0.20	0.23	101	99
	BQTerrace	0.23	-0.11	-1.27	102	103
Class C	BasketballDrill	0.40	0.35	0.30	101	100
	BQMall	0.18	0.05	0.00	101	99
	PartyScene	0.28	0.31	0.28	102	98
	RaceHorses	0.35	0.40	0.49	101	102
Class D	BasketballPass	0.10	-0.25	0.19	101	97
	BQSquare	0.06	0.34	-0.41	103	100
	BlowingBubbles	0.34	0.20	0.35	102	100
	RaceHorses	0.09	0.09	0.13	102	98
Overall	All	0.29	0.19	0.09	101	99

하여 Intel i7-5960X 3GHz CPU, 32GB 메모리의 동일한 성능을 갖는 5대의 PC로 클러스터를 구성하여 모든 실험을 진행하였다.

표 1은 BMS-2.0.1 대비 제안하는 방법 1과 2를 모두 사용할 경우의 BD-rate 성능 및 부호화 속도를 나타낸다. 제안하는 방법은 BMS-2.0.1 대비 Y, U, V 각각 0.29%, 0.23%, 0.04%의 BD-rate^[12] 성능을 보이고 거의 동일한 부/복호화 시간을 갖는다.

표 2는 BMS-2.0.1 대비 제안하는 방법 1의 BD-rate 성능 및 부호화 속도를 나타낸다. 제안하는 방법은 BMS-2.0.1 대비 Y, U, V 각각 0.29%, 0.19%, 0.09%의 BD-rate 성능을 보이고 거의 동일한 부/복호화 시간을 갖는다.

표 3은 BMS-2.0.1 대비 제안하는 방법 2의 BD-rate 성능 및 부호화 속도를 나타낸다. 제안하는 방법은 BMS-2.0.1 대비 Y, U, V 각각 0.1%, 0.15%, 0.15%의 BD-rate 성능을 보이고 거의 동일한 부/복호화 시간을 갖는다.

Cactus 영상과 BasketballDrill 영상과 같이 객체의 그림자로 인한 조명 변화가 심하거나, RitualDance 영상과 같이 자연광으로 인한 심한 국소적, 전역적 조명변화가 존재하는 경우 그렇지 않은 영상보다 상대적으로 LIC가 선택되는 비율이 높다. 따라서 LIC의 선택 비율이 높은 영상들이 그렇지 않은 영상보다 제안하는 방법에 의한 BD-rate 손실이 크다.

표 4는 기존의 LIC와 제안하는 LIC의 파라미터 유도 과

표 4. 양방향 예측에서 BMS2.0.1의 LIC와 제안하는 LIC의 파라미터 유도 과정의 복잡도 비교

Table 4. Comparison of complexity of parameter derivation process of LIC of BMS2.0.1 and proposed LIC in bidirectional prediction

	LIC in BMS2.0.1 in bi-prediction	Proposed LIC in bi-prediction
Multiplication	$2 \times 4 \times N$	0
Addition	$2 \times 4 \times N$	$4 \times N$

정의 복잡도 비교를 나타내고 있다. 기존의 LIC는 식 (1)와 같이 선형 모델을 사용하여 예측 블록에 밝기를 보상한다. 밝기 보상 파라미터로 scaling factor와 offset을 사용하고 식 (2)와 같이 현재 블록과 예측 블록의 주변 픽셀들을 사용하여 최소 자승법을 통해 유도한다. scaling 파라미터를 유도한 뒤 그 값을 사용하여 offset 값을 유도한다. 최소 자승법으로 scaling 파라미터를 유도하는 과정에서 현재 블록과 참조 블록의 주변 픽셀들에 대하여 곱셈과 덧셈이 각 $4 \times N$ (현재 블록의 너비와 높이 중 작은 값) 만큼 수행된다. 유도된 scaling값을 이용하여 곱해진 참조 블록의 주변 픽셀 값과 현재 블록의 주변 픽셀 값의 차이가 최소가 되도록 offset값을 유도한다. 따라서 offset을 유도하는 과정에서도 곱셈 연산이 수행된다. 양방향 예측인 경우 파라미터 유도에 사용되는 곱셈과 덧셈이 두 배로 증가하여 각각의 계산복잡도가 $2 \times 4 \times N$ 이 된다.

제안하는 LIC는 밝기 보상에 offset 파라미터만을 사용하기 때문에 식 (5)와 같이 간소화된 방법으로 파라미터를 유

표 3. BMS2.0.1(with LIC on) 대비 제안하는 방법 2를 사용한 경우의 BD-Rate 성능 및 부호화 시간

Table 3. BD-Rate performance and coding time when using proposed method 2 compared to BMS2.0.1(with LIC on)

Class	Sequence	BD-rate			Enc Time(%)	Dec Time(%)
		Y (%)	U (%)	V (%)		
Class B	MarketPlace	0.25	0.25%	0.22%	100	99
	RitualDance	0.18	0.19%	0.23%	100	99
	Cactus	0.20	0.26%	0.08%	100	99
	BasketballDrive	0.07	0.13%	0.20%	100	98
	BQTerrace	0.15	0.48%	0.09%	100	103
Class C	BasketballDrill	0.15	0.14%	0.12%	100	100
	BQMall	0.05	0.12%	0.20%	100	99
	PartyScene	0.06	0.10%	0.13%	101	98
	RaceHorses	0.22	0.40%	0.19%	100	102
Class D	BasketballPass	0.01	-0.37%	0.15%	100	97
	BQSquare	-0.01	0.10%	-0.46%	100	100
	BlowingBubbles	0.01	-0.15%	-0.01%	99	100
	RaceHorses	0.01	0.34%	0.81%	102	98
Overall	All	0.1	0.15	0.15	100	99

도한다. scaling 파라미터와 offset을 유도하기 위해 사용되었던 곱셈 연산을 수행하지 않아도 되고 덧셈 연산의 수행도 절반으로 감소하였다.

V. 결 론

차세대 압축 표준에 LIC가 추가되어 부/복호화기의 복잡도가 증가하였다. 양방향 예측일 경우 단방향 예측과 비교하였을 때 많은 정보를 사용하여 예측을 수행하여 압축 효율이 더 높지만 복잡도도 더 높다. 따라서 본 논문은 양방향 예측 모드에서 LIC의 복잡도 감소를 위해 양방향 예측일 경우 LIC를 간소화하여 수행하는 방법을 제안하였다. 양방향 예측일 경우 움직임 보상이 적용된 두 개의 예측 블록을 평균 내어 생성한 예측 블록에 LIC를 적용하고 LIC 파라미터로 오프셋 값만 사용하는 방법을 제안하였다. BMS-2.0.1 대비 Y, U, V 각각 0.29%, 0.23%, 0.04%의 BD-rate 성능을 보이고 부/복호화 시간은 비슷하였다. 부호화 효율의 감소가 있었지만, LIC 파라미터 유도과정에서 곱셈 연산이 제거되고 덧셈 연산이 절반으로 감소됨에 따라 계산복잡도가 크게 감소되었다. 본 논문에서 제안한 방법을 통해 LIC의 계산복잡도가 감소되었지만 일부 영상에 대하여 큰 부호화 효율 감소가 있었다. 이를 개선하기 위하여 제안하는 방법을 수행함에 있어서 영상의 특성을 고려하여 부호화 효율 감소를 완화하는 방법에 대한 연구가 필요하다.

참 고 문 헌 (References)

- [1] B. Bross, J. Chen, S. Liu, JVET-K1001, "Versatile Video Coding (Draft 2)", Jul. 2018.
- [2] VTM, https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM
- [3] BMS, https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_BMS
- [4] S. Park, D. Sim, B. Jeon, "Distributed Multi-view Video Coding Based on Illumination Compensation", The Institute of Electronics Engineers of Korea-SP, 45(6), 17-26, Nov. 2008.
- [5] ITU-T Recommendation H.264 and ISO/IEC 14496-10, "Advanced video coding for generic audiovisual services", May. 2003.
- [6] Sullivan, G. J., Ohm, J. R., Han, W. J., and Wiegand, T., "Overview of the high efficiency video coding(HEVC) standard", IEEE Transactions on circuits and systems for video technology, 22(12), 1649-1668, Dec. 2012.
- [7] J. M. Boyce, "Weighted prediction in the H. 264/MPEG AVC video coding standard." Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 International Symposium on. Vol. 3. IEEE, May. 2004.
- [8] J. Hur, S. Cho and Y. Lee, "Adaptive local illumination change compensation method for H. 264/AVC-based multiview video coding", IEEE Transactions on Circuits and Systems for Video Technology, 17(11), 1496-1505, Nov. 2007.
- [9] J. Chen, E. Alshina, G. J. Sullivan, J.-R. Ohm and J. Boyce, JVET-G1001, "Algorithm description of Joint Exploration Test Model 7", Jul. 2017.
- [10] H. Liu, Y. Chen, J. Chen, L. Zhang, M. Karczewicz, VCEG-AZ06, "Local Illumination Compensation", June. 2015.
- [11] F. Bossen, J. Boyce, X. Li, V. Seregin, K. Sühring, "JVET common test conditions and software reference configurations for SDR video", Jul. 2018.
- [12] G. Bjøntegaard, VCEG-M33, "Calculation of average PSNR differences between RD-curves", Apr. 2014.

저 자 소 개



최 한 술

- 2018년 2월 : 광운대학교 컴퓨터공학과 학사
- ORCID : <https://orcid.org/0000-0001-7947-0723>
- 주관심분야 : 영상신호처리, 영상압축, 컴퓨터비전

저 자 소 개



변 주 형

- 2019년 2월 : 광운대학교 컴퓨터공학과 학사
- ORCID : <https://orcid.org/0000-0002-6165-9189>
- 주관심분야 : 영상신호처리, 영상압축, 컴퓨터비전



방 건

- 2014년 2월 : 고려대학교 컴퓨터학과 박사 졸업
- 2002년 10월 ~ 2006년 2월 : ATSC T3/S2 ACAP 표준화 참여
- 2011년 9월 ~ 2012년 10월 : MIT RLE ATSP group 방문 연구원
- 2000년 4월 ~ 현재 : 한국전자통신연구원 실감AV연구그룹 책임연구원
- ORCID : <https://orcid.org/0000-0003-4355-599X>
- 주관심분야 : 영상압축, 영상신호처리, 컴퓨터비전, 인공지능, 실감미디어응용



심 동 규

- 1993년 2월 : 서강대학교 전자공학과 공학사
- 1995년 2월 : 서강대학교 전자공학과 공학석사
- 1999년 2월 : 서강대학교 전자공학과 공학박사
- 1999년 3월 ~ 2000년 8월 : 현대전자 선임연구원
- 2000년 9월 ~ 2002년 3월 : 바로비전 선임연구원
- 2002년 4월 ~ 2005년 2월 : University of Washington Senior research engineer
- 2005년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 교수
- ORCID : <http://orcid.org/0000-0002-2794-9932>
- 주관심분야 : 영상신호처리, 영상압축, 컴퓨터비전