

## 인트라 블록의 예측 정확도 향상 기술

정혜선, 강제원  
이화여자대학교 전자전기공학과, 스마트팩토리융합전공  
qjofc565@gmail.com, jewonk@ewha.ac.kr

### Improvements of Intra-predicted Block

Hyesun Jung, Je-Won Kang  
Department of Electronic and Electrical Engineering, Ewha Womans University  
Smart Factory Multidisciplinary Program, Ewha Womans University

#### 요약

본 논문은 딥러닝을 이용하여 예측 블록을 개선하는 화면 내 예측 기법을 제안한다. 컨볼루션 신경망 네트워크로부터 기존의 VVC의 화면 내 예측 모드를 통해 구성된 예측 블록과 주변 참조 샘플을 통과하여 보다 원본에 가까운 예측 블록을 생성한다. 따라서 예측 후 신호는 원본 블록과의 차분 신호를 줄여 비디오 부호화 성능을 향상하게 된다. 실험 결과, VTM-10.0 대비 휘도 성분에 대해 약 1.16%의 BD-rate를 개선하였다.

#### 1. 서론

최근 늘어나는 비디오 시청 수요에 따라 더 높은 품질의 비디오가 제작되고 있다. 영상 데이터의 용량이 더욱 커짐에 따라 비디오 압축 기술이 고도화 되고 있다. 비디오 압축에 관한 국제표준 그룹인 MPEG(Moving Picture Experts Group)에서 개발한 가장 최신의 국제 비디오 압축 표준은 VVC(Versatile Video Coding)/H.266[2]이다. VVC 기술에서는 향상된 기술로 HEVC(High Efficiency Video Coding)/H.265 [1] 대비 약 35%의 높은 부호화 효율을 제공한다.

VVC의 화면 내 예측은 기존 방식과 유사하게 현재 블록 주변의 공간적으로 인접한 샘플들을 활용하여 예측을 수행한다. DC 및 Planar의 무방향성 예측 모드와 65개의 방향성 예측 모드를 이용한다. 그 외에 새롭게 추가된 화면 내 예측 기술로 이전 HEVC에 비해 더 정교한 예측을 수행하게 되었다. 그럼에도 불구하고 화면 내 예측 방법은 복잡한 콘텐츠의 예측 블록을 정교하게 생성하기 어려운 문제가 있다.

CNN 기반의 기존 화면 내 예측 방식으로 [3]이 있다. [3]은 HEVC의 예측 블록과 그 주변의 참조 샘플을 입력으로 사용하여 예측 블록의 정확도를 올리는 기술을 제안하였다. 이 밖에도 이미지 인페인팅을 통해 예측 블록을 생성하고자 하는 연구가 있었다 [4].

본 논문에서는 VVC의 화면 내 예측 과정에서 컨볼루션 신경망 네트워크(convolutional neural network, CNN)를 이용하여 예측 블록의 정확도를 향상하여 부호화 효율을 개선하는 기술을 제안한다. All-Intra 코딩 시나리오에서 기존 부호화 방법 대비 제안 방법의 성능 향상을 실험적으로 보인다.

#### 2. 제안 방법

본 논문에서는 CNN을 활용하여 현재 예측 블록과 그 주변 참조 샘플을 입력으로 예측 블록을 개선해 화면 내 예측의 부호화 효율을 향상하는 것을 목표로 한다. 제안하는 네트워크는 3x3 컨볼루션으로 구성하는 네트워크를 이용한다. 그리고 deconvolution layer를 연결하여 구성한다. 이 때 입력은  $p \times p$  블록 주위 경계에 왼쪽 및 위의 참조 샘플  $r$ 을 포함한  $(p+r) \times (p+r)$ 의 블록이고 출력은 예측 정확도가 향상된  $p \times p$  블록이다. 그림1은 제안 네트워크 구조를 보인다.

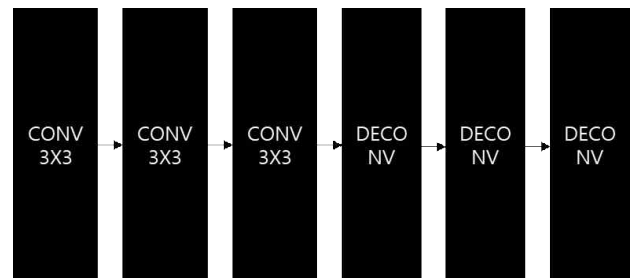


그림1. 제안하는 CNN 기반 입력 블록 개선 네트워크

네트워크를 학습하기 위한 손실함수는 원본 블록  $X$ 와 개선된 예측 블록  $X'$  사이의 차분 신호를 discrete cosine transform하여 나온 변환 계수들의 합을 통해 만든다. 식 1은 손실 함수를 보인다.

$$L = |K(X - X')K^T|, \dots (1)$$

식 (1)에서  $K$  는 DCT 변환 행렬이다. 식(1)의 학습 과정은 차분 신호의 요소인 변환 계수값들이 0이 되도록 학습이 된다.

블록의 개선 여부는 coding unit (CU) 단위마다 시그널링 되는 하나의 플래그를 이용하여 예측 블록을 개선 여부를 비트율 최적화를 통해 결정하게 된다.

### 3. 실험 결과

본 논문에서 제안하는 알고리즘의 성능을 평가하기 위하여 VVC 참조 소프트웨어인 VTM 10.0 [5] 에서 제안하는 방법의 성능을 비교하였다. 실험은 All-Intra(AI) 코딩 시나리오를 사용하여 QP 22, 27, 32, 37에서 BD-rate를 측정하였다.

실험 결과, 기존 부호화 방식 대비 제안하는 기법에 의한 부호화 방식은 평균적으로 약 1.16%의 BD-rate 감소 효율을 보였다.

표1. 제안 방법의 BD-rate(%) 비교

Classes	Sequence	BD-Rate (Y)
Class B	Kimono	-0.4%
	ParkScene	-1.0%
	Cactus	-1.0%
	BasketballDrive	0.5%
	BQTerrace	-1.3%
	Average	-0.7%
Class C	BaskektballDrill	-1.7%
	BQMall	-1.8%
	PartyScene	-1.7%
	RaceHorsesC	-1.0%
	Average	-1.5%
Class D	BasketballPass	-0.7%
	BQSquare	-1.9%
	BlowingBubbles	-1.9%
	RaceHorses	-2.0%
	Average	-1.6%
Class E	FourPeople	-0.6%
	Johnny	-1.1%
	KristenAndSara	-0.8%
	Average	-0.9%
Overall Average		-1.16%

### 4. 결론

본 논문은 딥러닝을 이용하여 예측 블록을 개선하는 화면 내 예측 기법을 제안한다. 컨볼루션 신경망 네트워크로부터 기존의 VVC의 화면 내 예측 모드를 통해 구성된 예측 블록과 주변 참조 샘플을 통과하여 보다 원본에 가까운 예측 블록을 생성한다. 따라서 예측 후 신호는 원본 블록과의 차분 신호를 줄여 비디오 부호화 성능을 향상하게 된다. 실험 결과,

VTM-10.0 대비 휘도 성분 에 대해 약 1.16%의 BD-rate을 개선하였다.

## ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2022R1A2C4002052).

## 참조문헌

- [1] Sullivan, Gary J., et al. "Overview of the high efficiency video coding (HEVC) standard." IEEE Transactions on circuits and systems for video technology 22.12 (2012): 1649-1668.
- [2] B. Bross, J. Chen, J.-R. Ohm, G. J. Sullivan, and Y.-K. Wang, "Developments in international video coding standardization after AVC, with an overview of versatile video coding (VVC)," Proceedings of the IEEE, 2021
- [3] Y. Wang, et al. "Multi-scale convolutional neural network-based intra prediction for video coding." IEEE Transactions on Circuits and Systems for Video Technology 30.7 (2019): 1803-1815.
- [4] L. Zhu, et al. "Generative adversarial network-based intra prediction for video coding." IEEE transactions on multimedia 22.1 (2019): 45-58.
- [5] Versatile Video Coding Test Model (VTM), 10.0 [Online], "[https://vcgit.hhi.fraunhofer.de/jvet/VVCSsoftwareVTM/-/releases/VTM-10.0.](https://vcgit.hhi.fraunhofer.de/jvet/VVCSsoftwareVTM/-/releases/VTM-10.0)"