정지영상 압축을 위한 인공신경망 내 비선형 변환 계층 분석

이주영, 조승현, 김휘용, 최진수
한국전자통신연구원
leejy1003@etri.re.kr, shcho@etri.re.kr, hykim5@etri.re.kr, jschoi@etri.re.kr

A study on nonlinear transform layers in neural networks for image compression

Jooyoung Lee, Seunghyun Cho, Hui Yong Kim, Jin Soo Choi
ETRI

요 약

인공신경망의 확산 및 보급에 따라 적용 영역이 확대되고 있으며 여러 분야에서 획기적인 성능 향상을 이루고 있다. 영상 압축 분야의 기술개발은 기존 코덱 구조 내 각 요소기술의 성능향상을 위한 인공신경망 기술 분야와 기존 코덱 구조가 아닌 end-to-end 학습을 통한 인공신경망 기반 기술 분야로 나뉘어 진행되고 있다. 본 논문에서는 end-to-end 학습을 통한 인공신경망 기술의 비선형 변환 계층 중 GDN(generalized divisive normalization) 계층이 영상 압축에 미치는 영향을 분석한다.

1. 서론

점차 인공신경망의 확산 및 보급에 따라 적용 영역이 확대되고 있으며 영상 및 컴퓨터 비전, 음성 처리 및 인식 등 여러 분야에서 획기적인 성능 향상을 이루고 있다. 특히 사물 인식 등 풀어야 할 문제가 직관적인 소위 one second rule[1][1]이 적용되는 분야의 경우, 인공신경망의 목적함수를 비교적 명확하고 빠르게 설계할 수 있으므로 인공신경망 기술의 보급 및 적용이 빠르게 이루어졌다. 반면 영상 압축 분야의 경우 여타의 분야와는 달리 풀어야 할 문제가 비교적 직관적이지 않아 타 분야 대비 인공신경망 기술의 적용이 비교적 더딘 편이었으나, 최근 정지영상 압축을 중심으로 기술 개발이 시작 단계에 있다.

영상압축 분야의 인공신경망 기반 기술개발은 기존 코덱 구조 내 각 요소기술의 성능향상을 위한 기술 분야와 기존 코덱 구조가 아닌 인공신경망의 end-to-end 학습을 통한 기술 분야로 나뉘어 진행되고 있다. 기존 코덱 기반 분야는 인공신경망이 강점을 보이는 분류 기술, AR(artifact reduction) 기술, 프레임 예측 기술 등을 기반으로 기존 코덱을 구성하는 개별 요소기술의 성능을 향상시키는 방법이며, end-to-end 학습을 통한 인공신경망 기술은 기존 코덱처럼 변환 기술, 예측 기술, 양자화 기술을 개별적으로 구분하지 않고, 해당 기술을 아우른 인공신경망이 R-D(rate-distortion) 측면에서 최적화된 결과를 도출하도록 학습하는 것이다. 대표적으로 Balle[2,3], Theis[4] 등이 있으며 현재 state-of-art 기술인 [3]은 BPG 에 근접한 성능을 보이고 있다. Balle 은 [2,3]에서 네트워크 구성을 위한 비선형 레이어로 그동안 보편적으로 많이 사용되어 왔던 ReLU, leaky ReLU, sigmoid, hyperbolic tangent 등의 방식이 아닌 GDN(generalized divisive normalization) [5]을 이용하였다. GDN 은 영상 내 혼재하는 다양한 분포의 데이터를 가우시안 형태로 변환하거나, 혹은 역변환 할 수 있는 정규화 기술로 데이터 간의 상호 의존성(correlation)을 효과적으로 제거할 수 있다. Balle 은 [6]에서 비선형 레이어로 GDN 을 이용하는 경우 정지영상 압축 측면에서 효과가 있음을 보였다. 그러나 인공신경망의 특징 중 하나인 레이어의 수가 증가할수록 복잡한 비선형 분포를 표현할 수 있는 점을 고려할 경우, [6]에서 사용된 인공신경망은 비교적 적은 수의 레이어로 구성되어 있으므로, GDN 이 범용적으로 우수한 성능을 보이는지에 대한 근거가 부족하다.

본 논문에서는 end-to-end 학습을 통한 인공신경망 기술의 비선형 변환 계층 중 GDN(generalized divisive normalization) 계층이 영상 압축에 미치는 영향을 분석한다. 구체적으로, 인공신경망이 적은 수의 레이어로 구성된 경우와 비교적 많은 수의 레이어로 구성된 경우에 대해 GDN 이용이 압축성능에 미치는 영향을 분석한다.

2. 영상 압축을 위한 end-to-end 학습 기법

본 논문에서 비선형 레이어의 성능을 비교하기 위해 [2,4]와 유사한 방식을 이용한다. [2]는 정지영상 압축을 위한 인공신경망의 end-to-end 학습기법으로, 비트량과 왜곡에 대한 두 개의 항으로 목적함수를 구성한다. 왜곡의 경우 범용적으로 널리 사용되는 MSE(mean square error) 방식을 이용하며, 비트량을 담당하기 위한 목적함수 항은 다음과 같이 구성한다. 인코더 네트워크의 출력 은닉변수를 z 라고 할 때, z 의 확률분포

[1] 일반적인 사람이 1 초 이내에 해결할 수 있는 문제를 의미

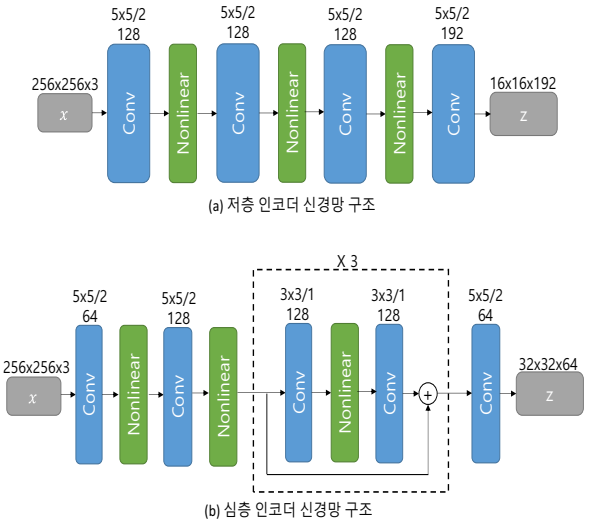


그림 1 검증용 인코더 신경망 구조

p(z)는 분석적으로(analytically) 계산할 수 없다. 이는 인코더 네트워크를 이용하여 p(z|x)가 주어지더라도, p(z)는 식 (1)과 같으므로, 식이 매우 복잡해질 뿐만 아니라 모든 x 에 대한 계산을 매 학습 시마다 수행할 수 없기 때문이다. 따라서 본 논문에서는 한정된 모델 파라미터로 표현 가능한 q(z)를 p(z)의 근사 모델로 이용하여 z 의 엔트로피 추정치를 계산한다. [2]의 경우 히스토그램 방식과 유사한 비모수적(non-parametic) 모델을 q(z)로 이용하였으나, 식 (1)에서 z|x는 확률변수이므로, p(z)는 중심극한정리에 의해 가우시안 분포에 가까워짐을 감안하여 본 논문에서 편의상 모수적(parametic) 가우시안 분포를 q(z)로 이용하였으며, 평균값은 0 으로 가정하고 표준편차는 학습을 통해 도출하였다.

$$p(z) = \frac{1}{N}\sum_{i} p(z|x_i) \qquad (1)$$

상기 언급한 바와 같이, 본 논문에서는 실제 분포 p(z) 대신 q(z)를 이용하여 z 의 엔트로피를 계산하였으므로, 이는 크로스 엔트로피이며, 식으로 나타내면 식 (2)와 같다.

$$\begin{aligned} H(p,q) &= E_p[-\log_2 q(z_i)] \\ &= -\frac{1}{N}\sum_{i} \log_2 q(z_i) \\ &= H(p) + D_{KL}(p|q) \end{aligned} \qquad (2)$$

즉, 식 (2)의 크로스 엔트로피 H(p,q)는 z 의 실제 분포 p 대신 근사 분포 q 를 이용하여 계산된 엔트로피를 의미한다. 따라서 H(p,q)는 은닉변수 z 의 실제 엔트로피와, p 가 아닌 q 를 이용함에 따라 증가하는 정보량인 KL-divergence 의 합이 된다. 즉 [2,4]의 방식에서 R-D 최적화는 상기 H(p,q)를 최소화하는 방향으로 진행되므로 H(p)와 KL-divergence 를 모두 최소화하는 방향으로 학습이 진행된다. 여기서 H(p)의 최소화는 인코더 인공신경망이 가급적 낮은 엔트로피의 z 를 출력하도록 하는 역할을 하며, KL-divergence 의 최소화는 q 의 모델 파라미터가 최대한 p 를 잘 근사하도록 하는 역할을 한다. 결론적으로 식 (3)과 같이 비트율 최소화를 담당하는 크로스 엔트로피 기반의 항과 왜곡 최소화를 담당하는 MSE 기반의 목적함수 항인 $d(x,\hat{x})$의 가중치 조절을 통해 다양한 비트율의 정지영상 압축 신경망을 학습할 수 있다.

$$L = -\frac{1}{N}\sum_{i} \log_2 q(z_i) + \lambda\, d(x,\hat{x}) \qquad (3)$$

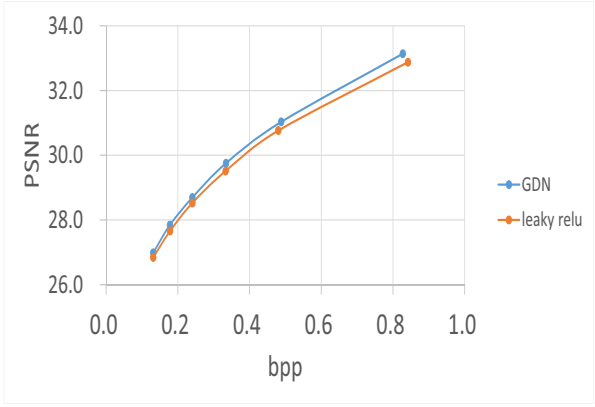
식 (3)에서 z 는 라운딩된 이산 값으로, 라운딩 지점에서 미분이 불가능한 문제가 발생하며, 이는 [2]의 방식에 따라 균등분포 랜덤 노이즈를 추가하여 미분 가능한 형태로 근사하여 문제를 해결하였다. 본 논문에서는 앞서 언급한 바와 같이 [2]와 달리 q(z)를 간소화하여 0 를 평균으로 하는 가우시안 모델을 이용하였으며, 공간적으로(spationally) 동일한 피쳐를 학습하는 컨볼루셔널(convolutional) 오토인코더임을 감안하여, 은닉변수 내 하나의 채널당 하나의 가우시안 분포 모델을 이용하였다.

3. 검증용 인공신경망 구조

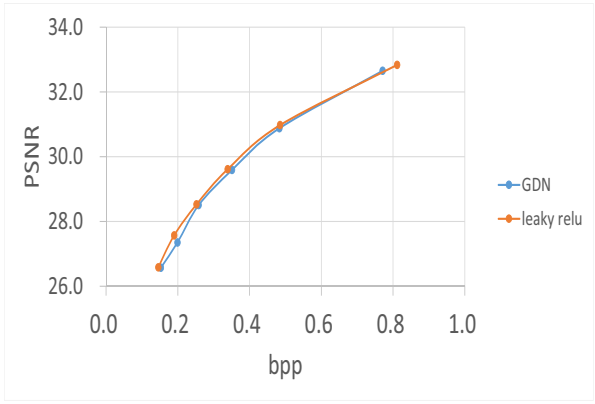
[6]에서는 2 장의 방식대로 최적화된 오토인코더의 각 계층에서 비선형 변환을 위한 레이어로 GDN 을 이용하는 경우, 범용적으로 널리 사용되는 ReLU, leaky ReLU, sigmoid, hyperbolic tangent 등의 비선형 레이어에 비해 정지영상 압축 측면에서 효과가 있음을 보였다. 본 논문에서는 GDN 의 범용성을 검증하기 위해 비교적 간단한 저층 구조의 오토인코더와 상대적으로 깊은 계층으로 구성된 오토인코더를 이용하였으며, 각 실험에 사용된 신경망의 구조는 그림 1 과 같다. 그림 1 에서 저층 인코더 구조는 [2]의 구조와 유사하며, 심층 인코더 구조는 [4]에서 사용된 구조와 유사하다. 저층 인코더의 경우 4 개의 컨볼루셔널 계층과 3 개의 비선형 변환 계층으로 구성하였으며, 심층 인코더 구조는 9 개의 컨볼루셔널 계층과 5 개의 비선형 계층으로 구성하였다. 각 인코더에 대응하는 디코더 또한 각각 [2] 및 [4]와 동일한 구조를 활용하였다. GDN 방식의 성능을 검증하기 위한 비선형 변환 계층 대조군으로 leaky ReLU 변환을 이용하였으며, 저층 환경과 심층 환경에서 각각의 정지영상 압축 성능을 측정하여, GDN 방식의 효용성을 검증하였다.

4. 실험 및 검증

3 장에서 제시한 저층 및 심층 인공신경망의 압축 성능에 각 비선형 레이어가 미치는 영향을 비교하기 위해, tensorflow 를 이용하여 네트워크를 구축하였으며, 은닉변수의 엔트로피 코딩을 위해 공개된 산술 코딩(arithmetic coding) 코드[7]를 수정/이용하였다. 각 엔트로피 코딩 시의 심볼 확률은 2 장에 기술한 바와 같이 학습된 각 채널 별 모델을 이용하였다. 학습셋은 YFCC100M [8] 이미지 데이터셋으로부터 랜덤 추출한 163,798 개의 128x128 패치를 이용하였으며, 8 개의 랜덤패치로 구성된 배치에 대해 총 500,000 회 학습을 수행하였다. 초기 학습율은 0.0001 의 값을 이용하였으며, 300,000 회부터



(a) 저층 인코더 신경망



(b) 심층 인코더 신경망

그림 2 계층 심도 및 비선형 변환 방식 별 R-D 곡선

50,000 회마다 1/2 의 비율로 학습율을 경감하였다. 학습 최적화 방식은 ADAM[9]을 이용하였다. 테스트셋으로 KODAK 이미지셋[10]을 이용하였으며, 계층 심도 및 비선형 변환 방식 별로 각각 6 단계의 λ값을 이용하여 네트워크를 학습시킨 후, 각 네트워크 별 압축 파일의 비트량과 복원 결과이미지의 PSNR 평균값을 측정하였다.

그림 2 는 상기 방식에 따라 학습된 각 네트워크의 비트율 및 PSNR 을 나타낸 그래프이다. 그림 2(a)는 저층 인코더 신경망의 결과를 나타낸 것으로, GDN 이 leaky ReLU 비선형 변환 방식에 비해 우수한 압축 성능을 보인다. 이는 [6]에서 Balle 이 제시한 결과와 유사하며, 레이어의 개수가 제한된 환경에서 GDN 이 각 레이어의 데이터 분포를 비선형적으로 정규화 하는 방식이 단순한 leaky ReLU 방식 대비 효율적임을 알 수 있다. 반면, 그림 2(b)와 같이 신경망의 계층 수가 높은 경우 GDN 과 leaky ReLU 방식의 압축 성능은 매우 유사해진다. 이는 ReLU, leaky ReLU 등 간단한 비선형 변환을 이용하더라도 레이어의 수가 깊어질 경우 다양한 분포를 잘 표현하는 심층 신경망의 특징에 부합한다. 즉, 본 실험 결과, 신경망의 계층이 깊어질 경우에는 단순한 비선형 변환으로도 압축을 위한 데이터 분포를 GDN 과 동등한 수준으로 표현할 수 있음을 유추할 수 있다. 따라서, GDN 은 보다 적은 계층으로 데이터의 분포를 잘 표현하는 효율성을 제공하나, 다양한 심층신경망 환경에서 범용적인 우수성을 제공한다고 볼 수 없다.

5. 결론

본 논문에서는 end-to-end 학습을 통한 인공신경망 기술의 비선형 변환 계층 중 GDN(generalized divisive normalization) 계층이 영상 압축에 미치는 영향을 분석하였다. 신경망을 구성하는 계층의 수가 적은 경우 GDN 은 단순한 leaky ReLU 비선형 변환 방식에 비해 우수한 압축 성능을 보였으나, 계층의 수가 깊은 경우 두 비선형 변환 방식은 거의 동등한 성능을 보였다. 이는 ReLU, leaky ReLU 등 간단한 비선형 변환을 이용하더라도 레이어의 수가 깊어질 경우 다양한 분포를 잘 표현하는 심층 신경망의 특징에 기인한 것으로 사료된다. 즉, GDN 은 적은 계층으로 데이터를 표현하는 경우 우수한 압축 효율성을 제공하나, 다양한 심층신경망 환경에서 범용적인 우수성을 제공하지 못할 것으로 사료된다.

감사의 글

이 논문은 2018 년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (2017-0-00072, 초실감 테라미디어를 위한 AV 부호화 및 LF 미디어 원천기술 개발)

참고문헌

[1] https://hbr.org/2016/11/what-artificial-intelligence-can-and-cant-do-right-now

[2] J. Ballé, V. Laparra, E. P. Simoncelli, "End-to-end optimized image compression", Proc. Int. Conf. Learn. Representat, pp. 1-27, 2017.

[3] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, “Variational image compression with a scale hyperprior,” 6th Int. Conf. on Learning Representations, 2018, accepted. [Online]. Available: https://openreview.net/forum?id=rkcQFMZRb.

[4] L. Theis, W. Shi, A. Cunningham, and F. Huszar, “Lossy image compression with compressive autoencoders,” International Conference on Learning Representations, 2017.

[5] Ballé, Johannes, Valero Laparra, and Eero P. Simoncelli (2016a). “Density Modeling of Images Using a Generalized Normalization Transformation”. In: arXiv e-prints. Presented at the 4th Int. Conf. on Learning Representations. arXiv: 1511.06281.

[6] “Efficient Nonlinear Transforms for Lossy Image Compression”, [Online]. Available: https://arxiv.org/abs/1802.00847.

[7] https://github.com/nayuki/Reference-arithmetic-coding

[8] https://webscope.sandbox.yahoo.com/catalog.php?datatype=i&did=67&guccounter=1

[9] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In ICLR, 2015.

[10] E. Kodak. Kodak lossless true color image suite (PhotoCD PCD0992).

CNN 기반의 VVC 인-루프 필터 설계

문현철, 김재곤
한국항공대학교
hcmoon@kau.kr, jgkim@kau.ac.kr

CNN Based In-loop Filter in Versatile Video Coding (VVC)

Hyeonchul Moon and Jae-Gon Kim
Korea Aerospace University

요 약

본 논문에서는 새로이 시작된 비디오 압축 표준인 VVC(Versatile Video Coding)의 인-루프(in-loop) 필터링을 위한 CNN 구조를 제안한다. 제안하는 CNN 구조는 복호화된 영상을 입력으로 하고 원본 영상과 복호화된 영상의 오차를 손실함수로 사용하여 학습을 진행한다. 또한, 비디오 부호화에서의 다양한 크기의 CU(Coding Unit)를 고려한 다양한 크기의 컨볼루션 필터를 사용하여 특징을 추출하는 구조에 기반하고 있다. 실험을 통하여 제안한 CNN 기반의 필터링이 VVC 의 시험모델인 VTM(VVC Test Model)의 인-루프 필터링의 성능을 개선할 수 있음을 확인하였다.

1. 서론

기존 HEVC 의 2 배의 압축 성능을 목표로 한 초고효율의 새로운 비디오 부호화 표준인 VVC(Versatile Video Coding) 표준화가 시작되었다. VVC 후보 기술로 새로운 부호화 툴과 함께 화면내 예측, 화면간 예측, 인-루프필터[2], [3], 변환 부호화 등에 딥러닝을 적용한 기술들도 제안되었다[1]-[3].

딥러닝 기술들은 영상인식/분류, 객체검출 등 다양한 분야에 뛰어난 성능을 보여주고 있으며, 그 중에 CNN (Convolutional Neural Network) 알고리즘은 필터링 기능과 영상복원에 특화된 특징을 가지고 있다. 이 알고리즘을 응용한 SRCNN(Super-Resolution CNN) 구조는 저해상도의 영상을 고해상도로 복원한다. SRCNN 은 기존의 영상처리 기반의 SR 기법보다 우수한 성능을 가지며 다양한 응용에서 기본 구조로 채택되고 있다[4], [5]. 또한 기존의 SRCNN 구조를 개선하여 CNN 계층을 더 깊게 구성한 VDSR(Very Deep Super-Resolution)[6], 영상 압축의 아티팩트(artifact)를 줄이는 ARCNN(Artifact Reduction CNN)[7], 다수의 필터를 적용하도록 구성한 VRCNN(Variable-filter-size Residual-learning CNN)[8] 등이 있다.

본 논문에서는 필터링 기능과 잡음 제거에 용이한 CNN 의 특징을 고려하여 비디오 압축의 블록 경계에서의 아티팩트와 양자화 잡음을 제거하기 위한 인-루프 필터에 적용하는 방안을 제시한다.

본 논문의 구성은 다음과 같다. 2 절에서는 본 논문에서 제안하는 인-루프 필터에 적용할 CNN 구조를 기술하고, 3 절에서는 기존의 기법과 제안하는 기법의 실험결과 및 성능비교를 제시한다. 마지막으로, 4 절에서는 결론을 맺는다.

2. 제안하는 CNN 구조

제안하는 CNN 기반의 인-루프 필터의 구조는 SRCNN 및 VRCNN 구조를 기반으로 한다. 그림 1 의 제안하는 구조의 입력영상은 복호화된 영상을 입력으로 한다. 학습과정에서는 출력영상을 압축전의 원본영상과 입력영상의 차이인 잔차영상을 출력하게 유도하고, 최종 출력영상은 입력영상과 잔차영상을 더한 영상으로 정의한다. 이 최종 출력영상과 입력영상의 평균최소자승오차(MSE: Mean Squared Error)를 손실함수(loss function)로 학습을 진행한다. 제안한 구조의 CNN 기반의 인-루프 필터를 적용할 때는 출력영상과 입력영상을 더한 최종 결과영상을 얻는다.

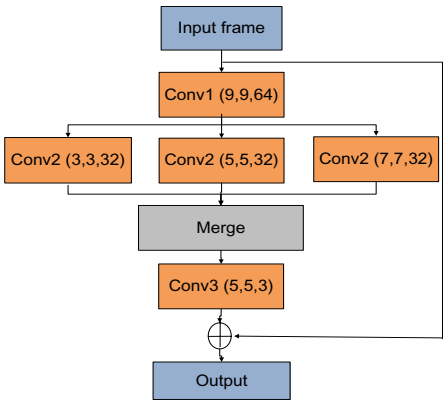


그림 1. 제안하는 CNN 구조

그림 1 의 제안하는 CNN 구조는 식 (1) ~ (3)으로 표현된다. 괄호의 숫자는 (필터사이즈, 필터사이즈, 특징맵의 수)를 의미하며, Conv1, Conv2 층은 Relu 활성화 함수를 사용한다. 식 (1)에서 Wi 와 Bi 는 각각 i 계층에서의 가중치와 Biases 를 의미한다. 즉, 한 층의 출력 값은 전 층의 출력 값과 현재 컨볼루

션 계층의 가중치를 곱한 값에 현재 층의 Biases 를 더한 값이 되며, 이를 활성화 함수(Relu)를 통과한 값이 다음 계층의 입력 값으로 된다. 식 (2)는 그림 1 의 Merge 층을 나타낸 것으로, Conv2 층의 출력 값이 각각 3, 5, 7 의 필터 크기를 사용하는 Conv2 층의 평균 값으로 설정됨을 나타낸 것이다. 이와 같이 제안한 CNN 구조는 다양한 컨볼루션 필터 크기를 사용하고 이를 다시 합치는 방식이다. 식 (3)은 마지막 층의 출력 Y 에 대한 식이며, 이 때는 활성화 함수 Relu 를 사용하지 않고 최종 결과물인 잔차영상을 생성한다.

$$F_i = \max(0, W_i * F_{i-1} + B_i), i = 1, 2 \quad (1)$$

$$F_2 = Average(F_{2,3} + F_{2,5} + F_{2,7}) \quad (2)$$

$$Y = W_3 * F_2 + B_3 \quad (3)$$

3. 실험 결과

본 논문에서 제안하는 CNN 기반의 인-루프 필터의 성능을 확인하기 위한 실험은 All Intra 모드에서 6 개의 시퀀스 BasketballDrill, BQMall, PartyScene, BlowingBubbles, BQSquare 그리고 PeopleOnstreet 를 학습 데이터를 사용했으며, 다양한 QP 의 값(27, 32, 37)으로 부호화한 시퀀스를 혼합하여 학습을 진행하였다. 테스트 시퀀스로는 테스트 시퀀스와 겹치지 않도록 BasketballDrill, BQMall 의 각각의 1~100 번째 프레임을 사용하였다. 실험결과는 VVC(Versatile Video Coding)의 참조 SW 코덱인 VTM-1.0 의 결과를 기준으로 비교하였다.

표 1. 제안 기법의 성능평균 PSNR

방법	PSNR(dB)
인-루프 적용 전	32.74
인-루프 적용 후	33.10
VRCNN [5]	33.32
제안하는 기법 (CNN)	33.34

표 1 의 실험결과는 각 경우에서의 실험에 사용한 4 개의 QP 22, 27, 32, 37 에서의 평균 PSNR 을 나타낸 것으로, 제안하는 CNN 기법은 VCC 의 인-루프 필터 대비 평균 0.24dB 가 향상됨을 확인할 수 있었으며, 필터 크기를 다양하게 적용할 수 있는 기존의 VRCNN 과 비교했을 때 약 0.02dB 의 향상을 확인할 수 있다.

표 2. 각 QP 값에 따른 제안한 기법 성능(PSNR 이득)

QP	제안 기법
22	+0.03
27	+0.17
32	+0.22
37	+0.35

표 2 의 결과와 같이 높은 QP 에서 상대적으로 제안 기법의 성능 개선이 큰 것을 확인할 수 있다.

그림 2 는 기존의 기법과 제안된 기법의 주관적 화질을 비교한 것으로 CNN 기반의 제안 기법이 보다 개선된 화질을 얻을 수 있음을 확인할 수 있다.

기존기법 제안기법

그림 2. 주관적 화질 비교(BasketballDrill, 6 번째 프레임)

4. 결론

본 논문에서는 CNN 기반의 VVC 인-루프 필터를 제시하였다. 다양한 크기의 CU 를 고려하여 다양한 크기의 필터를 포함한 CNN 구조를 사용하였다. 실험결과 기존의 VCC 의 인-루프 필터와 비교했을 때 0.24 dB PSNR 이득을 얻을 수 있었고, 주관적 화질 비교에서도 두드러진 화질 개선을 확인할 수 있었다.

제안 기법은 다양한 QP 에서 성능개선을 보이지만 낮은 QP 에 대해서는 상대적으로 개선 정도가 낮으며 이에 대한 개선이 필요한다. 또한, BD-Rate 의 성능과 보다 다양한 시퀀스에서의 실험을 통한 성능 확인과 Random Access 및 Low Delay 의 부호화 모드에서의 성능을 확인할 예정이다. 또한, 향후 JVET 에 제안된 VVC 의 인-루프 필터와 성능 비교을 진행할 예정이다.

감사의 글

이 논문은 본 연구는 산업통상자원부 국가표준기술원에서 시행한 국가표준기술력향상사업[10084981, 인공지능 기반의 패턴인식 기술 국제표준화 개발]의 지원을 받아 수행된 연구임.

참 고 문 헌

[1] S. Liu et al, “JVET AHG report: Neural Networks in Video Coding,” JVET document, JVET-J0008, Apr. 2018.

[2] C. W. Hsu, et al, “Description of SDR video coding technology proposal by MediaTek,” JVET document, JVET-J0018, Apr. 2018.

[3] Zhou et al, “Convolutional Neural Network Filter (CNNF) for intra frame,” JVET document, JVET-I0022, Apr. 2018.

[4] W. S. Park and M. Kim, "CNN-based in-loop filtering for coding efficiency improvement," In. Proc. IVMSP 2016, July 2016.

[5] C. Dong, C. C. Loy, K. He, K., and X. Tang, “Image super-resolution using deep convolutional networks,” IEEE Trans. Pattern Analysis Machine Intelligence, vol. 38, no. 2, Feb. 2016.

[6] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks,” In Proc. CVPR 2016, Jan. 2016.

[7] Y. Dai, D. Liu, and F. Wu, "A Convolutional neural network approach for post-processing in HEVC intra coding," arXiv:1703.03502, 2017.

[8] Ke YU, et al. “Deep convolution networks for compression artifacts reduction,” arXiv:1608.02778, 2016.

DenseNet 기반의 이미지 압축

박운성, 김문철
한국과학기술원
pys5309@kaist.ac.kr, mkim@ee.kaist.ac.kr

DenseNet based Image Compression

Woonsung Park Munchurl Kim
KAIST

요 약

본 논문에서는 기존 신경망 기반의 이미지 압축에 많이 사용되었던 신경망인 ResNet 을 대신하여 더 적은 개수의 파라미터를 사용하여 좋은 성능을 낼 수 있는 신경망 구조인 DenseNet 을 이미지 압축에 사용한다. 이미지 압축을 위해 사용되는 신경망 구조는 일반적으로 오토 인코더 구조인데, 병목 층에서 정보 손실이 상당히 많이 발생한다. 따라서 이미지 압축에서 신경망 내에서의 정보 전달은 상당히 중요하다. 기존의 논문에서는 이를 위해 이전의 정보를 그대로 뒤로 전달해주는 구조인 ResNet 을 사용하여 깊은 층에 대해서도 수렴이 잘 되는 결과를 보여주었다. 그러나 많은 수의 파라미터를 사용하는 단점을 해결하기 위해 본 논문에서는 DenseNet 을 이미지 압축에 사용하였고, 병목 층에서의 정보 손실로 인해 이미지의 고주파수 성분이 사라지는 현상을 해결하기 위해 원래 이미지와 JPEG2000 으로 압축한 이미지와의 차이를 추가 입력으로 넣어주어서 주관적인 화질을 개선하였다.

1. 서론

최근 들어 기계학습이나 신경망을 기반으로 한 영상처리 기법이 상당히 많이 나오고 있다. 먼저 다양한 구조의 신경망을 이미지 초해상화[1, 2]에 적용하여 성능을 향상시키는 연구가 활발히 이루어지고 있다. 이외에도 디노이징[3], 압축으로 인한 노이즈 제거[4] 등과 같은 이미지 복원과 관련한 영상 처리 기술에도 신경망이 적용되고 있다. 이러한 영상 처리 기술들의 공통점은 왜곡된 영상들을 복원하거나 화질을 향상시키는 목적으로 신경망을 사용했다는 것이다.

고전적인 이미지 압축은 크게 JPEG[5], JPEG2000[6]을 예시로 들 수 있다. 이들은 기본적으로 특정한 종류의 선형 변환을 통해 픽셀들 간의 상관관계를 줄이고, 양자화를 통해 필요한 정보량을 줄이게 된다. 이후에 다시 역선형변환을 통해 이미지를 복구한다. 즉, 양자화 과정에서 정보 손실이 발생하기 때문에 이를 최소화 하는 선형 변환을 찾는 것이 중요하고, 기존의 전통적인 이미지 압축에서는 선형 변환이나 양자화 과정을 독립적으로 최적화하였다. 그러나 실제로 선형 변환이 정보를 압축하는 데에 있어서 이상적인 변환이 아닐 수 있고, 선형 변환 과정과 양자화 과정 그리고 역선형변환 과정을 통합하여 압축을 최적화할 필요성이 있다.

최근에 딥러닝 기술의 발전과 기존 전통적인 이미지 압축의 한계를 발전시키기 위해 신경망을 이용한 이미지 압축에 관한 연구가 활발히 진행되고 있다. 초기에는 단순한 구조의 컨볼루션 신경망을 사용하여 이미지 압축에 적용하기도 하였고[7], 이후에는 RNN, LSTM[8]과 같은 복잡한 구조의 신경망을 사용하여 이미지 압축의 성능을 향상 시켰다[9]. 또한 상당히 깊은 층의 신경망을 학습시키기 위해 ResNet[10]과 같은 구조를 사용하여 이미지 압축에 적용한 연구[11]도 있다. 그러나 초기의 신경망 기반의 이미지 압축은 성능이 기존의 이미지 압축 기술보다 떨어지거나 신경망의 구조가 너무 복잡하다는 단점이 있다.

본 논문에서는 이러한 문제점을 해결하기 위해 파라미터를 줄이는 데에 효과적인 신경망 구조를 이미지 압축에 적용하는 방법을 제안한다. 구체적으로 이전의 층을 참조하여 다음 층을 만들어내는 신경망을 사용하여 적은 파라미터에도 효과적으로 정보가 전달될 수 있도록 하였다.

본 논문의 구성은 다음과 같다. 2 절에서는 제안하는 DenseNet 기반의 이미지 압축에 사용한 신경망 구조를 설명하고, 3 절에서는 이러한 구조를 이미지 압축에 적용하여 향상된 성능을 실험을 통해서 확인한다. 마지막으로 4 절에서는 본 논문에 대한 결론을 맺는다.

2. 본론

일반적으로 신경망 기반의 이미지 압축은 오토 인코더 구조를 기반으로 한다. 오토 인코더는 입력과 출력이 같도록 신경망이 구성되어 있고, 신경망 사이에는 입력보다 작은 차원의 특징맵으로 매핑되는 병목 층이 존재한다. 이러한 구조의 특성은 병목 층에서 정보가 줄어든 것을 원래의 입력에 가장 가깝도록 복원하는 것이 이미지 압축의 과정으로 볼 수 있다. 또한 기존의 이미지 압축은 선형 변환을 기반이지만, 신경망을 사용하게 되면 각 층 사이에 비선형 함수가 포함되어

있어서 비선형 변환과 같은 효과를 얻을 수 있다. 일반적으로 이미지의 특성은 단순한 선형 변환으로는 정보를 최적으로 줄일 수 없기 때문에 신경망을 통해 비선형 변환을 사용하여 정보를 더 압축할 수 있도록 해준다.

하지만 이미지 압축을 위한 신경망 구조는 깊은 층으로 구성된 신경망으로 이루어져 있어야하는데, 단순한 컨볼루션 신경망으로 층을 쌓으면 쉽게 학습되지 않는다. 특히 중간에 존재하는 병목 층으로 인해 정보가 중간에 손실되기 때문에 원래 입력 이미지에 가까운 이미지로 출력하는 것이 어렵다. 그래서 최근에 이전의 특징맵을 나중의 특징맵과 더하는 식의 신경망 구조인 ResNet 을 사용하여 위의 문제점을 어느 정도 해결하였다. 그러나 이를 위해서는 많은 수의 채널 수가 각 층마다 필요하여 신경망을 학습할 때 필요한 파라미터 수가 상당히 많아진다.

제안하는 DenseNet 기반의 이미지 압축은 신경망 구조로 DenseNet[12]을 사용한다. DenseNet 은 이미지 분류에 사용하기 위해 처음 제안된 신경망 구조로, 기존의 ResNet 을 조금 더 발전시킨 형태의 신경망 구조이다. 기존의 ResNet 은 이전의 특징맵에서 어느 정도 거리가 있는 특징맵과의 스킵 커넥션을 통해 학습과 정보 전달이 잘 이루어지도록 하였다. DenseNet 은 이에 더하여 각 층마다 이후의 모든 층에 특징맵 정보가 전달 될 수 있도록 모든 층과의 스킵 커넥션이 포함되어 있다. 따라서 상대적으로 더 적은 개수의 채널을 사용하여도 깊은 층의 신경망을 학습할 수 있고, 정보 전달이 잘 이루어질 수 있다. 본 논문은 적은 파라미터를 사용하여도 정보 전달이 잘 이루어질 수 있는 DenseNet 을 기반으로 한 이미지 압축 방법을 제안하였다.

본 논문에서 제안한 DenseNet 기반의 이미지 압축 구조는 오토 인코더의 인코더 부분과 디코더 부분으로 나눌 수 있다. 먼저 인코더 부분을 그림 1 에 묘사하였다.

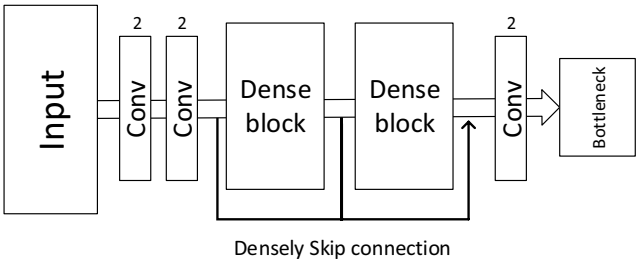


그림 1. DenseNet 기반의 이미지 압축 구조의 인코더 부분.

제안하는 DenseNet 기반의 이미지 압축 구조의 인코더의 입력으로 이미지가 들어오면 두 개의 컨볼루션 신경망을 통해 크기가 1/4 로 줄어들게 된다. 이후의 특징맵은 Dense 블록을 통해 더 높은 레벨의 특징맵을 추출해내고, 마지막으로 크기를 1/2 로 줄이는 컨볼루션 신경망을 통과시켜서 병목 층의 특징맵을 만들어낸다. 그래서 병목 층 이전의 신경망 구조를 비선형 변환에 해당하는 부분이라고 생각할 수 있다. 병목 층에서는 적절한 양자화가 이루어지고, 양자화된 병목 층의 특징맵은 디코더 부분을 통과하게 된다.

이러한 인코더 구조에서 추가 실험으로 진행한 부분은 추가 입력으로 원본 이미지와 JPEG2000 으로 압축한 이미지와의 차이를 넣어주는 것이다. 보통 원본 이미지와 JPEG2000 으로 압축한 이미지의 차이는 고주파수 성분이 주를 이루게 된다. 신경망 기반의 이미지 압축도 마찬가지로 고주파수 성분이 많이 사라지게 된다. 따라서 이를 해결하기 위해 본 논문에서는 입력 이미지 이외에 추가입력으로 고주파수 성분에 대한 정보를 넣어주어서 고주파수 성분에 대한 복원이 더 잘 이루어지도록 하였다. 제안한 이미지 압축 구조의 디코더 부분은 그림 2 에 묘사하였다.

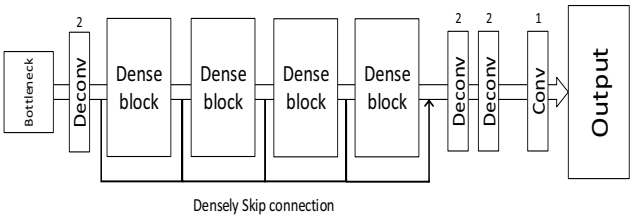


그림 2. DenseNet 기반의 이미지 압축 구조의 디코더 부분.

제안하는 DenseNet 기반의 이미지 압축 구조의 디코더의 입력으로 양자화된 병목 층의 특징맵이 들어오면 컨볼루션 신경망을 통해 크기가 2 배로 늘어나게 된다. 이후의 특징맵은 Dense 블록을 통해 복원을 위한 더 높은 레벨의 특징맵을 추출해낸다. 이 때, 인코더의 Dense 블록보다 더 많은 4 개의 Dense 블록을 사용한 이유는 이미지 복원을 위해서는 더 복잡한 구조의 신경망이 필요하기 때문이다. 최종적으로 크기를 2 배로 늘리는 컨볼루션 신경망을 2 개 통과시킨 후 컨볼루션 신경망으로 최종 복원을 하면 결과 이미지가 생성된다. 이를 입력 이미지에 가깝도록 MSE 관점에서 학습을 하였다.

그림 1, 2 에서 표기된 Dense 블록의 자세한 구조는 그림 3 에 묘사하였다.

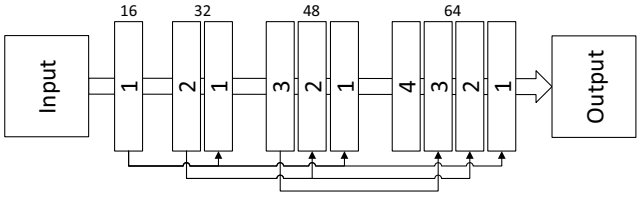


그림 3. 이미지 압축에 사용된 Dense 블록.

이 실험에서 사용된 Dense 블록은 위와 같이 각 층의 특징맵이 이후의 특징맵에 모두 채널에 쌓여서 연결이 된다. 기존의 ResNet 과는 달리 각 특징맵이 이후의 특징 맵에 모두 영향을 주고, 단순히 값을 더하는 것이 아닌 특징 맵 자체를 뒤로 전달하게 된다. 이런 특성 때문에 상대적으로 ResNet 에 비해 더 적은 채널 수를 사용하여 학습할 수 있다.

최종적으로 그림 1, 2, 3 에서 제시한 신경망 구조를 사용하여 입력 이미지가 들어오면, 결과 이미지로 같은 이미지가 나오도록 학습을 시키게 된다. 이 때 중간의 병목 층의 특징맵 크기를 어느 정도로 하느냐 그리고 양자화를 어느 정도로 하느냐에 따라 압축의 정도가 결정된다.

3. 실험 결과

본 논문에서 제안한 DenseNet 기반의 이미지 압축의 성능을 ResNet 기반의 이미지 압축의 성능과 비교하였다. 이 때 각 신경망 구조의 파라미터 수는 약 34 만 개를 사용하도록 채널 수를 조절하였다. 학습 이미지는 ETH Zurich 에서 이미지 압축 Challenge 를 위해 제공했던 이미지 중에서 585 개의 2k 이미지를 사용했다. 테스트 이미지는 마찬가지로 ETH

Zurich 에서 제공한 41 개의 2k 이미지[13]를 사용하였다. 배치 크기는 32 개의 128×128 을 사용하였고, 컨볼루션 필터 크기는 크기를 줄일 때는 4×4, DenseNet 에서의 컨볼루션 필터 크기는 3×3 을 사용하였다. 비선형 함수는 ReLu 를 사용하였고, 모든 네트워크는 학습할 때 Adam Optimizer[14]를 사용하였다. 최종적으로 입력 이미지와 출력 이미지와의 MSE 를 100 epoch 동안 학습시켰다. 그리고 학습이나 테스트는 모두 NVIDIA Titan XP GPU 를 통해 진행되었다.

본 실험에 사용된 네트워크는 총 3 개이다. 먼저 본 논문에서 제안한 DenseNet 기반의 일반적인 이미지 압축 네트워크이고, 두 번째는 추가 입력으로 JPEG2000 으로 압축한 이미지와 원본 이미지와의 차이를 넣어주는 네트워크이다. 마지막으로 비교를 위해 비슷한 개수의 파라미터를 사용한 ResNet 기반의 이미지 압축 네트워크이다. 이들의 성능을 같은 압축률로 압축했을 때의 41 개의 테스트 이미지에 대한 PSNR, MS-SSIM[15] 평균값으로 비교를 하였다.

표 1. 세 가지 네트워크의 압축 성능 비교

	PSNR (dB)	MS-SSIM
ResNet	28.86	0.9497
DenseNet	**29.15**	0.9502
DenseNet2	28.98	**0.9509**

서로 다른 네트워크의 공정한 비교를 위해 같은 수의 파라미터로 신경망을 구성하였다. 때문에 ResNet 은 기존의 논문에서 제시했던 채널 수보다 더 적게 구성하였고, 이를 학습했을 때 표 1 의 결과처럼 학습을 완벽하게 하지 못했다. 그러나 DenseNet 의 경우 비슷한 파라미터를 사용했음에도 같은 epoch 에 대해 학습을 상당히 잘 한 것을 확인할 수 있다. 이는 상대적으로 더 적은 채널을 가지고 정보 전달을 잘 할 수 있는 DenseNet 의 구조가 도움이 되었다고 생각한다. DenseNet2 의 경우 추가 입력을 사용한 네트워크인데, 이 네트워크의 결과는 PSNR 관점에서는 추가 입력을 사용하지 않은 네트워크보다 안좋지만, 주관적인 화질을 더 잘 측정할 수 있는 MS-SSIM 관점에서 측정한 결과 더 좋은 성능을 보였다. 이는 압축으로 인해 사라진 고주파수 성분만 따로 추가 입력으로 보내면 이를 병목 층에서의 특징맵에 반영할 수 있는 것으로 보인다.

4. 결론

본 논문에서는 DenseNet 기반의 이미지 압축 구조를 제안하여 기존 ResNet 기반의 이미지 압축보다 더 좋은 압축 성능을 보였다. 기존의 ResNet 기반의 이미지 압축은 학습을 위해 상대적으로 많은 파라미터를 사용해야만 했지만, 제안하는 DenseNet 기반의 이미지 압축은 각 층의 특징맵이 이후의 모든 층의 특징맵으로 연결되기 때문에 학습에 도움이 되고, 상대적으로 적은 채널을 사용하여 비슷한 성능을 얻을 수 있다. 또한 압축으로 잃어버린 고주파수 성분과 관련된 정보를 추가 입력으로 넣어주어서 주관적인 화질 성능을 올리도록 하였다. 추후의 연구는 GAN 을 활용한 주관적인 화질 성능 향상이 있고, 정지 영상이 아닌 동영상에 대한 압축 네트워크를 구성하는 것이다.

Acknowledgement

이 논문은 2018 년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No. 2017-0-00419, 스마트 방송 미디어를 위한 지능형 고실감 영상처리 연구).

참고문헌

[1] C. Dong, C.C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," *European Conference on Computer Vision*, pp. 184-199, 2014.

[2] J. Kim, J. Lee, and K. Lee, "Accurate image super-resolution using very deep convolutional networks," *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1646-1654, 2016.

[3] J. Xie, L. Xu, and E. chen, "Image denoising and inpainting with deep neural networks," *Advances in neural information processing systems*, pp. 341-349, 2012.

[4] C. Dong, Y. Deng, and C. Loy, "Compression artifacts reduction by a deep convolutional network," *Proceedings of the IEEE international Conference on Computer Vision*, pp. 576-584, 2015.

[5] G.K. Wallace, "The JPEG still picture compression standard," *IEEE transactions on consumer electronics*, 38(1), xviii-xxxiv, 1992.

[6] D. Taubman and M. Marcellin, "JPEG2000 image compression fundamentals, standards and practice: image compression fundamentals, standards and practice," Vol. 642, Springer Science & Business Media, 2012.

[7] J. Balle, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," arXiv preprint arXiv:1611.01704, 2016.

[8] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, 9(8), 1997.

[9] G. Toderici, D. Vincent, N. Johnston, S. J. Hwang, D. Minnen,J. Shor, and M. Convell, "Full resolution image compression with recurrent neural networks," arXiv preprint arXiv:1608.05148, 2016.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.

[11] L. Thesis, W. Shi, A. Cunningham, and F. Huszar, "Lossy image compression with compressive autoencoders," arXiv preprint arXiv:1703.00395, 2017.

[12] G. Huang, Z. Liu, "Densely connected convolutional networks," *Proc. of the IEEE conference on computer vision and pattern recognition*, Vol. 1, No. 2, pp. 3, 2017.

[13] http://www.compression.cc/

[14] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[15] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," *Signals, Systems and Computers*, Vol. 2, pp. 1398-1402, 2004.

딥러닝 기반 화면 간 예측 부호화 기법

*이정경, **강제원
이화여자대학교, 전자공학과
*jkl931204@naver.com, **jewonk@ewha.ac.kr

Deep Learning based Inter Prediction Coding Technique

*Jung Kyung Lee **Jewon Kang
Department of Electronic and Electrical Engineering, Ewha Womans University

요 약

본 논문에서는 비디오 부호화 과정 중 화면 간 예측 부호화 과정에 딥러닝을 적용하여 부호화 효율을 제고하는 알고리즘을 제안한다. 보다 구체적으로 딥러닝으로 생성한 가상의 픽쳐를 현재 프레임의 참조 픽쳐로 사용하는 방법에 대해 설명한다. 부호화 과정에서 복원된 픽쳐 두 장을 이용하여 가상의 보간 픽쳐를 생성하고 생성된 보간 픽쳐를 참조 프레임으로 사용하여 화면 간 예측의 효율을 높인다. 실험에 따르면 참조 픽쳐 리스트를 수정하여 참조 구조를 변경함으로써 HEVC 참조 코덱인 HM 16.9 대비 평균 1.4%의 BD-rate 감소 효율을 제공하였다.

1. 서론

최근 딥러닝 기반의 알고리즘을 이용하여 기존의 기술을 대체하거나 효율성을 높이려는 시도가 많아지고 있다 [3-5]. 그러나 화면 간 예측 부호화에 적용된 연구는 거의 시도되지 않았다. 본 논문에서는 High Efficiency Video Coding (HEVC) 부호화 과정에 딥 러닝 기반의 영상 보간 기술을 적용하여 새로운 참조 프레임을 생성함으로써 화면 간 예측 기법에 적용하는 방식을 제안한다.

2. 관련 연구

HEVC 의 임의 접근 모드 (Random Access)는 GOP (Group Of Pictures) 단위로 화면 내 예측만 수행하는 I-슬라이스를 주기적으로 넣고 B-슬라이스를 통해 단방향과 양방향 예측 모드를 선택하여 화면 간 예측 부호화를 수행한다. 현재의 픽쳐와 상관도가 높은 몇 개의 복원 픽쳐를 복호화 픽쳐 버퍼(DPB:Decoded Picture Buffer)에 저장하고, 참조 픽쳐로 사용하여 예측을 한다. 참조 픽쳐는 현재 픽쳐를 기준으로 디스플레이 순서에 따라 참조 픽쳐 리스트 0 과 참조 픽쳐 리스트 1 에 저장한다 [1]. 현재 픽쳐는 참조 픽쳐 리스트와 참조 픽쳐 인덱스를 이용하여 참조 중인 픽쳐를 알 수 있고, 해당 블록은 참조 픽쳐 리스트, 참조 픽쳐 인덱스, 그리고 움직임 벡터를 이용하여 예측 신호를 생성한다. 그림 1 은 HEVC 의 계층적 B 구조를 이용한 임의 접근 모드에서 두 방향의 참조 픽쳐 중 시간 인덱스 (temporal ID)를 고려하여 계층적 부호화를 수행하는 과정을 보인다 [2].

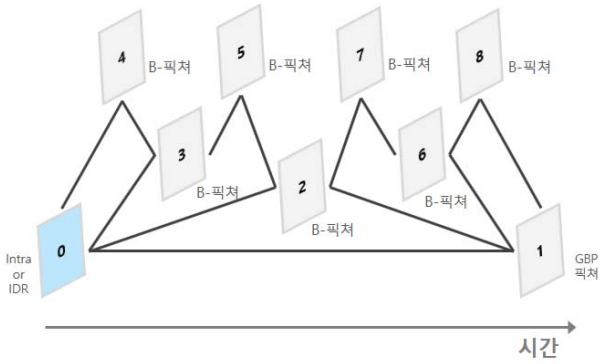


그림 1. HEVC 의 임의 접근 모드

HEVC 부호화에 뉴럴 네트워크를 적용하려는 시도가 있었다. 화면 내 예측에서 CU 의 모드 선택의 고속화를 위해 컨벌루션 네트워크를 사용하는 연구가 있다[3]. 네트워크 구조 중 하나인 Fully connected network 를 사용하여 화면 내 예측에서 주변의 다수의 참조 블록을 이용하여 현재 블록의 예측을 보다 정확하게 수행하는 연구가 제안되었다 [4]. 또한 인루프 필터에서 왜곡된 픽쳐를 복원하기 위한 알고리즘 대신 딥러닝 기반 네트워크를 사용한 연구가 있다 [5]. 그러나 화면 간 예측 기법에 딥 러닝 기법을 적용한 방식은 현재까지 거의 없었다.

3. 제안 방법

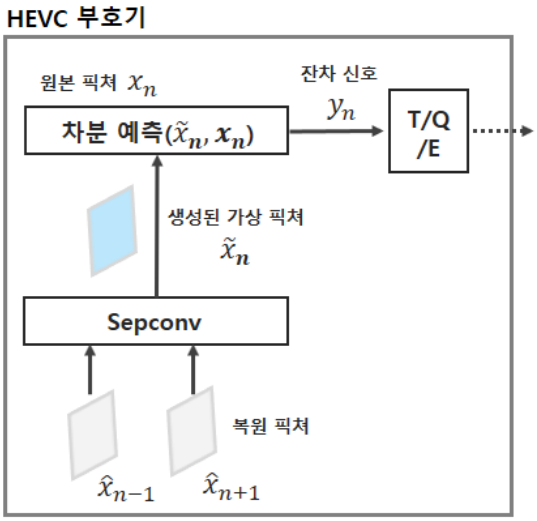


그림 2. 제안 알고리즘

그림 2 는 제안 알고리즘에서 HEVC 부호기에서 현재 픽쳐 이전에 부호화된 픽쳐으로부터 보간하여 생성한 가상 픽쳐를 화면 간 예측에 사용하며 부호화하는 과정을 보인다. 먼저 두 장의 복원 픽쳐를 딥러닝 네트워크에 입력으로 넣어 가상 픽쳐를 생성한다. 가상 픽쳐를 참조 프레임으로 사용하여 움직임 예측 및 보상에 사용한다. 복호 측에서도 같은 방식으로 참조 프레임 리스트를 구성하며 움직임 보상 후에 현재 프레임을 복원한다.

A. 딥러닝 기반 가상 픽쳐 생성

가상 픽쳐를 생성하기 위해 두 장의 입력 픽쳐에 대해 시간적으로 중간에 위치해 있는 보간 픽쳐를 생성하는 뉴럴 네트워크를 사용한다. 보간을 위한 여러 딥러닝 구조 중 Adaptive separable convolution network(Sepconv)[6] 네트워크를 사용한다. Sepconv 의 구조는 다른 네트워크에 비해 보간 픽쳐를 예측하는 성능도 뛰어나며 적은 수의 파라미터로 학습이 가능하다. 그림 3 처럼 부호화-복호화 구조로 이루어진 이 네트워크는 컨볼루션을 통해 특징을 추출하고 이 추출된 특징들은 마지막 단에 4 개의 하위 네트워크로 나누어져 각각의 1D 커널을 이용해 픽셀 기반 필터링을 수행한다. 결과적으로, 출력 이미지로 시간적으로 두 개의 입력 픽쳐의 가운데에 있는 이미지가 예측, 생성된다.

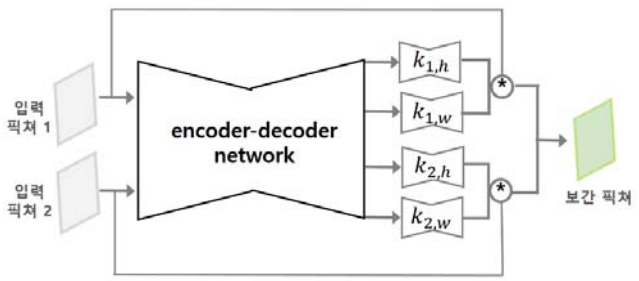


그림 3.
Adaptive separable convolution network 구조

Sepconv 에 두 장의 입력 픽쳐는 DPB 에 있는 픽쳐 중 현재 픽쳐와 가장 POC 차이가 작은 두 장을 사용한다. 기존 HEVC 코덱에서 사용한 참조 프레임 리스트 0 과 참조 프레임 리스트 1 에서 각각 한 장씩 사용한다. 참조 프레임 리스트 0 의 프레임을 첫번째 입력으로 넣고 참조 프레임 리스트 1 의 프레임이 두번째 입력으로 넣으면 한 장의 보간된 프레임을 얻게 된다. 이 과정을 입력의 순서를 바꾸어 보간된 프레임을 한 장 더 얻어 총 두 장의 보간 프레임을 생성한다. 따라서 현재 픽쳐에 대해 총 두 장의 딥러닝 기반 가상 픽쳐를 얻는다.

B. 참조 프레임 리스트 구성

HEVC 참조 코덱에서 지정된 참조 프레임 리스트 0 과 참조 프레임 리스트 1 에 채워진 첫 번째 픽쳐는 기존과 동일하게 유지한다. 각각의 리스트에 채워지는 두 번째 픽쳐는 생성한 가상 픽쳐를 삽입한다. 참조 픽쳐 리스트의 변경은 가장 마지막 레이어에 위치한 픽쳐에 대해서만 실행하였다.

Table 1. 참조 픽쳐 리스트의 구성

POC	참조 픽쳐 리스트 0		참조 픽쳐 리스트 1	
8	0	0	0	0
4	0	8	8	0
2	0	4	4	8
1	0	가상 픽쳐 1	2	가상 픽쳐 1
3	2	가상 픽쳐 3	4	가상 픽쳐 3
6	4	0	8	4
5	4	가상 픽쳐 5	6	가상 픽쳐 5
7	6	가상 픽쳐 7	7	가상 픽쳐 7

4. 실험 결과

실험 조건 본 논문에서 제안하는 HEVC 부호화 시 참조 픽쳐 리스트 구성 변경의 성능을 평가하기 위하여 HEVC 참조 소프트웨어인 HM 16.9 에서 공통 실험 조건 CTC (Common Test Condition)[7]을 참고하여, 제안하는 방법의 성능을 비교하였다. 실험은 HM 16.9 에서 GOP 의 수는 8 로 설정 후 실험을 진행하였다. Sepconv 모델은 별도의 재학습이 없이 기존 사전학습(pre-trained) 모델을 사용하였다.

실험 결과 기존 부호화 방식 대비 깊이 딥러닝 기반 참조 픽쳐 리스트 구성을 통한 부호화 방식은 평균적으로 약 1.4%의 BD-rate 감소 효율을 제공하였다.

그림 4 와 그림 5 를 보면 Partyscene 영상과 Johnny 영상의 원본 픽쳐와 비교 픽쳐 간의 차분 이미지를 통해 픽쳐 간의 오차를 확인 할 수 있다. 차분 픽쳐 1 은 원본 픽쳐와 복원 픽쳐 간의 차분 이미지이고 차분 픽쳐 2 는 원본 픽쳐와 딥러닝 네트워크를 통해 얻은 보간 픽쳐 간의 차이다. 차분 픽쳐 3 과 차분 픽쳐 4 는 가상 픽쳐를 생성할 때 사용했던 두 장의 픽쳐와 원본 픽쳐 간의 차분이다. 차분 픽쳐 2 와 차분 픽쳐 3,4 를 비교해서 보면 가상 픽쳐가 원본 픽쳐와 유사함을

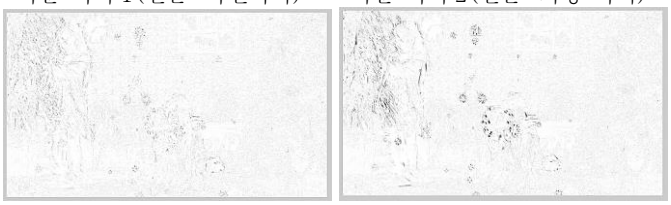
확인 할 수 있다.

Table 2 실험 결과

Class	Video	BD-rate		
		Y	U	V
416x240	RaceHorses	-1.7%	-0.8%	-1.2%
	BlowingBubbles	-2.8%	-1.7%	-2.0%
	BasketballPass	-3.4%	-2.3%	-3.9%
	BQSquare	-3.3%	-0.2%	-0.8%
832x480	RaceHorses	-0.5%	-0.2%	-0.6%
	BQMall	-2.5%	-1.4%	-1.7%
	PartyScene	-1.4%	-0.8%	-1.0%
	BasketballDrill	0.0%	0.0%	-1.0%
1920x1080	BasketballDrive	-0.1%	0.3%	0.6%
	BQTerrace	-0.6%	-0.4%	-0.4%
	Cactus	-1.6%	-0.3%	-1.2%
	Kimono1	-2.0%	1.0%	1.0%
	ParkScene	-2.0%	0.0%	0.0%
1280x720	Johnny	-0.1%	0.1%	0.2%
	FourPeople	-0.9%	-0.3%	-0.2%
	KristenAndSara	-0.3%	0.1%	0.1%
Average		-1.4%	-0.4%	-0.8%

원본 픽쳐 복원 픽쳐

차분 픽쳐 1(원본-복원픽쳐) 차분 픽쳐 2(원본-가상 픽쳐)



차분 픽쳐 3(원본-입력픽쳐 1) 차분 픽쳐 4(원본-입력픽쳐 2)

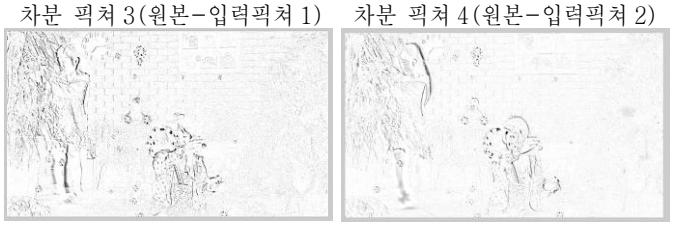
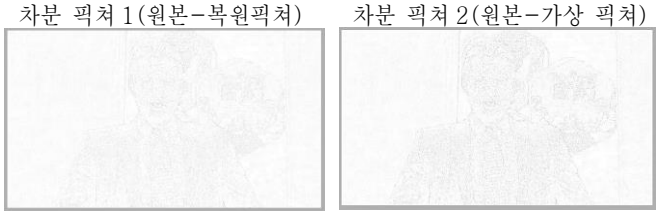


그림 4.
PartyScene 영상의 차분 픽쳐 비교

원본 픽쳐 복원 픽쳐

차분 픽쳐 1(원본-복원픽쳐) 차분 픽쳐 2(원본-가상 픽쳐)



차분 픽쳐 3(원본-입력픽쳐 1) 차분 픽쳐 4(원본-입력픽쳐 2)

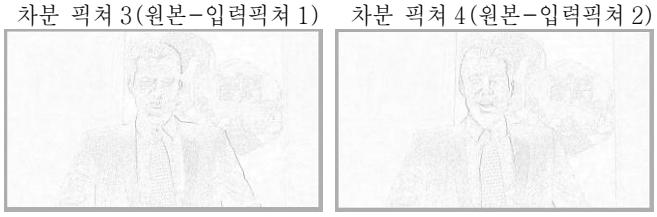


그림 5.
Johnny 영상의 차분 픽쳐 비교

5. 결론

본 논문에서는 딥 러닝 기반의 기술을 HEVC 부호화의 화면 간 예측에 적용하는 방법을 제안하였다. 제안 방법의 실험 결과, 기존 방법보다 평균 약 1.4%의 BD-rate 를 개선하였다. HEVC 부호화의 화면 간 예측 기법에 딥 러닝 기법을 적용한다는 점에 의의를 가진다.

감사의 글

This work was supported by Institute for Information and communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (2017-0-00072, Development of Audio/Video Coding and Light Field Media Fundamental Technologies for Ultra Realistic Tera-media)

5. 참고문헌

[1] G. J. Sullivan, J. M. Boyce, Y. Chen, J. R. Ohm, C. A. Segall and A. Vetro, "Standardized Extensions of High Efficiency Video Coding (HEVC)," in IEEE Journal of Selected Topics in Signal Processing, vol. 7, no. 6, pp. 1001-1016, Dec. 2013.

[2] R. Sjoberg et al., "Overview of HEVC High-Level Syntax and Reference Picture Management," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1858-1870,

Dec. 2012

[3] Z. Liu, X. Yu, Y. Gao, S. Chen, X. Ji and D. Wang, "CU Partition Mode Decision for HEVC Hardwired Intra Encoder Using Convolution Neural Network," in IEEE Transactions on Image Processing, vol. 25, no. 11, pp. 5088-5103, Nov. 2016.

[4] J. Li, B. Li, J. Xu and R. Xiong, "Intra prediction using fully connected network for video coding," 2017 IEEE International Conference on Image Processing (ICIP), Beijing, 2017, pp. 1-5.

[5] W. S. Park and M. Kim, "CNN-based in-loop filtering for coding efficiency improvement," 2016 IEEE 12th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), Bordeaux, 2016, pp. 1-5.

[6] Niklaus, Simon, Long Mai, and Feng Liu. "Video frame interpolation via adaptive separable convolution." arXiv preprint arXiv:1708.01692 (2017).

[7] F. Bossen, "Common test conditions and software reference config- urations," JCTVC-L1100, 12th JCT-VC meeting, Geneva, CH, Jan.2013.