특집논문 (Special Paper)
방송공학회논문지 제24권 제3호, 2019년 5월 (JBE Vol. 24, No. 3, May 2019)
https://doi.org/10.5909/JBE.2019.24.3.420
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

압축 영상 화질 개선을 위한 딥 러닝 연구에 대한 분석

이 영 운[a)], 김 병 규[a)‡]

Comparative Analysis of Deep Learning Researches for Compressed Video Quality Improvement

Young-Woon Lee[a)] and Byung-Gyu Kim[a)‡]

요 약

최근 CNN (Convolutional Neural Network) 기반의 화질 개선 기술이 H.265/HEVC와 같은 블록 기반 영상 압축 표준을 사용하여 압축된 영상의 화질을 향상시키는 데 적극적으로 사용되어 왔다. 이 논문은 이러한 영상 압축 기술을 위한 화질 개선 연구의 추세를 요약하고 분석하는 것을 목표로 한다. 먼저, 화질 개선을 위한 CNN의 구성 요소를 살펴보고 이미지 도메인에서의 사전 연구를 요약한다. 다음으로 네트워크 구조, 데이터셋 및 학습 방법의 세 가지 측면에서 관련 연구들을 정리하고 성능 비교를 위한 구현 및 실험 결과를 제시하고자 한다.

Abstract

Recently, researches using Convolutional Neural Network (CNN)-based approaches have been actively conducted to improve the reduced quality of compressed video using block-based video coding standards such as H.265/HEVC. This paper aims to summarize and analyze the network models in these quality enhancement studies. At first the detailed components of CNN for quality enhancement are overviewed and then we summarize prior studies in the image domain. Next, related studies are summarized in three aspects of network structure, dataset, and training methods, and present representative models implementation and experimental results for performance comparison.

Keyword : CNN, HEVC, Noise Reduction, Quality Enhancement

a)숙명여자대학교 IT공학과(Department of IT Engineering, Sookmyung Women's University)
‡ Corresponding Author : 김병규(Byung-Gyu Kim)
E-mail: bg.kim@sookmyung.ac.kr
Tel: +82-2-2077-7293
ORCID: https://orcid.org/0000-0001-6555-3464

※This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1A1B04934750).
· Manuscript received April 4, 2019; Revised May 10, 2019; Accepted May 10, 2019.

Ⅰ. 서 론

H.265/HEVC는 ITU-T VCEG 및 ISO/IEC MPEG에서 공동으로 개발한 영상 압축 표준으로서 현재 가장 널리 사용되고 있는 기술이다[1]. 이러한 종류의 블록 기반 영상 부호화 기술은 손실 압축 기법을 활용하여 목표 압축치를 달성하게 되는데 이로 인해 압축된 영상의 화질을 떨어뜨리는 다양한 노이즈 (블로킹 열화, 링잉 아티팩트 등)가 발생한다. 주관적/객관적 화질을 향상시키기 위해 영상 압축 표준에는 DF (de-blocking filter), SAO (sample adaptive offset) 등의 필터링 기술들이 포함되어 있으며 끊임없이 새로운 기술들이 많은 연구자들에 의해 제안되고 있다. 특히, 인공 지능 분야 (분류, 객체 인식 등)에서 높은 성과를 나타내고 있는 CNN (convolutional neural network)를 화질 향상에 적용하려는 시도가 많이 있었으며 계속적으로 의미있는 성과를 거두고 있다.

영상 부호화에 딥러닝 기법을 적용하는 목적은 크게 두 가지로 구분할 수 있다. 첫 번째는 영상 부호화의 고속화 또는 빠른 모드 결정 (fast mode decision)이다. 화면 내 예측 (intra prediction), 화면 간 예측 (inter prediction), 비트율 제어 (rate control), 변환 부호화 (transform coding) 등 부호화 표준을 구성하는 기술들은 율-왜곡 최적화 (Rate-Distortion Optimization: RDO)를 통해 최대의 압축률을 달성할 수 있는 최적의 모드를 결정하는 과정으로 볼 수 있다. 가능한 모든 모드를 비교하고 결정하는 과정은 과도한 연산량을 요구하므로 CNN 모델의 학습을 통해 최적화 계산을 간소화하고 부호화 효율을 높이는 방식이다.

두 번째는 화질 향상이다. 다양한 CNN 구조가 후처리 필터 또는 인트라-/인터-코딩을 위한 인-루프 필터 방식으로 제안되어 왔다. 최근 새로운 영상 부호화 기술을 표준화하기 위한 워킹 그룹에 CNN 기술 접목을 위한 애드-혹 그룹이 추가되는 등 활발한 연구 활동이 진행되고 있지만 여전히 많은 한계가 존재한다. 특히, 충분한 성능을 보장하기 위해 심층적인 네트워크를 추구하게 되면 과도한 계산량 및 메모리가 필요하게 되는데 이와 같은 비용에 비해 얻어지는 화질 개선량은 충분하지 않다.

단일 이미지와 달리 비디오는 공간적 중복성 뿐만 아니라 시간적 중복성으로 인해 복잡한 영상 특성을 갖는다. 그러므로 이를 적절히 반영할 수 있는 학습 모델을 고안하는 것이 중요한 문제이다.

본 논문은 딥러닝 특히, CNN을 사용하여 부호화 영상의 화질 향상을 위한 연구 동향을 분석하는 데 중점을 둔다. 2장은 화질 개선을 위한 CNN의 공통적인 구성 요소를 정리한다. 3장은 관련 연구로써 이미지 도메인에서의 이전 연구 결과들을 정리한다. 4장은 영상 부호화를 위해 제안된 대표적인 CNN 구조들을 네트워크 구조, 데이터셋, 학습 방법의 3가지 측면에서 요약한다. 5장은 성능 비교를 위한 구현과 실험 결과를 정리한다. 6장에서 분석 결과를 도출하고 7장에서 최종적인 결론을 맺는다.

Ⅱ. CNN 개요

부호화 영상의 화질 향상을 위해 노이즈가 존재하는 입력 이미지 또는 패치에 대해 노이즈가 없는 원본과 매핑하는 종단 간 접근법이 일반적으로 사용된다. 모델은 모델에 의해 추론된 이미지와 원본 사이의 차이 값을 손실함수로 정의함으로써 역 전파 방법에 의해 학습된다. CNN을 구성하는 구성 요소의 세부 사항은 다음과 같다.

$$(F*k)(p)=\sum_{s+t=p}F(s)k(t), \tag{1}$$

$$(F*_l k)(p)=\sum_{s+lt=p}F(s)k(t). \tag{2}$$

식 (1)은 일반적인 컨볼루션 연산을 나타내며 및 식 (2)는 팽창된 컨볼루션 (dilated convolution)을 나타낸다. F와 k는 각각 입력값과 컨볼루션 커널을 나타내며, l은 팽창 계수 (dilated factor)를 의미한다. 따라서 바이어스 b가 주어지면 $(F*k)(p)+b$ 또는 $(F*_l k)(p)+b$로 나타낼 수 있다. 팽창된 컨볼루션은 입력값에서 많은 전역적인 특징을 추출하기 위해 receptive field의 크기를 높이면서도 학습 매개변수를 늘리지 않는 방법이다.

$$ReLU(x)=\max(0,x). \tag{3}$$

$$PReLU(x)=\begin{cases}x, & \text{if } x>0,\\ ax, & otherwise.\end{cases} \tag{4}$$

식 (3)과 같이 컨볼루션 계층에 이어지는 활성화 함수로는 $ReLU$가 일반적으로 사용되지만 음수 값을 반영하여 정확성을 높이기 위해 식 (4)와 같이 $PReLU$가 사용되기도 한다. x는 입력값 즉, 컨볼루션 연산의 출력값을 나타내며 a는 학습 가능한 매개 변수이다. 만약, a가 고정된 상수 값일 경우 Leaky ReLU라고 칭한다.

$$\hat{x_i}\leftarrow\frac{x_i-\mu_B}{\sqrt{\sigma_B^2+\epsilon}}, \tag{5}$$

$$BN_{\gamma,\beta}(x_i)=\gamma\hat{x_i}+\beta. \tag{6}$$

식 (5)와 식 (6)은 배치 정규화 (batch normalization: BN) 수식을 나타내며, 모델에 따라 컨볼루션 계층에 포함된다. μ_B와 σ_B^2는 미니 배치의 평균과 분산이다. $\hat{x}$은 입력 x에 대한 정규화 된 값이고 ϵ은 상수 값이다. γ 및 β는 각각 스케일 및 시프트 계수를 의미한다. BN 기술은 모델의 훈련 과정에서 기울기가 소실되거나 발산하는 문제를 해결하기 위해 도입되었으며 $ReLU$와 같은 활성화 함수를 사용하거나 학습율을 조정하는 방법 등과 함께 사용된다.

$$x_L=x_l\sum_{i=l}^{L-1}F(x_i). \tag{7}$$

식 (7)은 잔차 네트워크 (Residual Network: ResNet)를 나타낸다. x_L은 특정 계층에서의 출력을 나타내며 x_l과 F는 해당 계층에 대한 입력과 연산을 각각 의미한다. F 함수 값과 계층의 입력값을 더하는 것은 스킵 연결 (skip connection)이라고 부르는데 이러한 잔차 학습 (residual learning)기술은 일반적으로 네트워크의 최초 입력과 최종 출력 사이에 스킵 연결을 추가하여 학습 속도와 정확도를 높이게 된다.

Ⅲ. 관련 연구

많은 연구 성과가 이미지 도메인에서의 성과로부터 시작되었으므로 이를 먼저 언급할 필요성이 있다. Dong[2]은 초해상도 (super-resolution) 문제를 풀기 위해 상대적으로 단순한 3 계층의 fully convolution network (FCN) 구조를 제안하였고 bicubic interpolation을 사용하여 확대된 이미지의 화질을 개선시켰다. 해당 연구를 통해 초해상도 분야에서 전통적으로 사용되던 스파스 코딩 (sparse coding) 기술이 CNN 구조로 대체될 수 있다는 사실을 입증하였다.

Dong의 모델은 초해상도 보다는 노이즈 감소 기술로 이해할 수도 있는데, 이는 이미 공간적으로 상향 조정된 이미지를 사용하여 학습하고, 보간법의 한계로 인해 발생되는 노이즈를 개선시켰다는 점 때문이다. Dong의 모델은 4 계층의 네트워크로 확장되어 JPEG 이미지에 대한 화질을 개선시키는 연구 결과를 내기도 하였다[3].

Kim[4]은 20 계층의 상대적으로 깊은 네트워크가 초해상도 문제 해결에 적합함을 보였다. 특히, 그들은 잔차 학습 기술과 gradient clipping과 같은 기술을 결합하여 깊은 네트워크를 짧은 시간에 학습시키는데 성공했다.

화질 향상을 위한 CNN 접근 방식은 3 가지 관점에서 분

표 1. CNN 구조 비교
Table 1. Comparison of CNN Structures

Author	# Layer	# Parameter	Convolution Layer Structure
Park	3	24,416	(9 × 9 × 64), (3 × 3 × 32), (5 × 5 × 1)
Wang	10	296,064	(3 × 3 × 64)×9, (3 × 3 × 1)
Dai	6	54,512	(5 × 5 × 64), (5 × 5 × 16), (3 × 3 × 32), (3 × 3 × 16), (1 × 1 × 32), (3 × 3 × 1)
Meng	75	126,336	CH1: (K1 × K1 × 28), (5 × 5 × 4)×4, (3 × 3 × 8)×4, (1 × 1 × 16)×4 CH2: (K2 × K2 × 28), (5 × 5 × 4)×8, (3 × 3 × 8)×8, (1 × 1 × 16)×8 CH3: (K3 × K3 × 28), (5 × 5 × 4)×12, (3 × 3 × 8)×12, (1 × 1 × 16)×12
Song	8	222,336 (127,350)	(3 × 3 × 64)×7, (3 × 3 × 1)

석할 수 있다. 첫째, 네트워크 구조의 차이를 분석할 수 있다. 그동안 제안되었던 네트워크는 3 계층의 상대적으로 얕은 네트워크에서 75 계층의 깊은 네트워크까지 다양하다. 대부분의 연구가 네트워크 각 계층의 구성을 컨볼루션과 활성함수만 사용하는 FCN 구조를 따르는 것으로 나타났다. 둘째는 데이터셋이다. 기본적으로 HEVC를 사용하여 부호화된 영상을 사용했다는 공통점이 있지만 입력 이미지 또는 패치 외에 추가적인 정보를 입력으로 사용한다거나 데이터를 수집하는 방식 등에서 차이가 있으며, 부호화에 사용된 참조 소프트웨어의 버전에서도 차이가 있다. 마지막으로, 학습 방법의 차이이다. 학습이란 모델의 학습 가능한 파라미터를 최적화시키는 문제이다. 이를 위해 다양한 역전파 알고리즘, 잔차 학습 및 gradient clipping 등과 같은 기법들이 적용 되었다.

4장부터 부호화 영상의 화질 개선을 위한 다섯 가지 CNN 연구를 선정하여 분석한다. 화질 개선에 관한 연구들은 Dong이 제안한 FCN 구조의 모델로부터 파생되었다고 볼 수 있다. 따라서, 본 논문의 범위는 공간적 (spatial) 특성만을 고려하여 제안된 FCN 구조들에 대해 분석하고자 한다.

생성모델 (generative model)과 판별 모델 (descriptive model)간의 경쟁 구조를 도입한 GAN 기반의 모델은 주관적 화질에 의존적인 성능을 보이므로 분석 범위에서 제외하였고 동영상에 적합한 시간적 (temporal) 특성에 초점을 맞춘 3D 컨볼루션 모델이나 Spatial-Temporal 모델 등은 본 논문의 분석 범위를 넘어선다.

Ⅳ. CNN 기반 화질 개선 기법 분석

1. 네트워크 구조

표 1은 제안된 각 네트워크들의 구조를 개략적으로 정리한 것으로 전체 계층 수, 가중치 파라미터의 수와 각 계층의 커널 구조를 나타내고 있다. 전체 계층 수는 한 개의 컨볼루션 연산이 포함된 경우 한 개의 계층으로 계산했다. 가중치 파라미터의 수는 바이어스를 제외한 컨볼루션 커널의 가중치 및 배치 정규화의 가중치만을 고려하여 계산되었다. 컨볼루션 커널의 구조는 계층 간의 연결 순서는 고려하지 않고 커널 크기와 특징 맵의 수만을 나타낸 것이다. 또한, 동일한 구조를 가진 반복된 계층에 대해서 위 첨자로 반복의 수를 나타내고 있다.

Park[5]은 Dong의 3 계층 모델에 잔차 학습 기법을 적용하였다[2]. 즉, 모델의 초기 입력 값은 최종 출력 값에 연결되어 전체 네트워크가 하나의 ResNet을 구성하게 된다. 제로 패딩 (zero padding)은 컨볼루션 연산에 의해 출력값이 공간적으로 축소되는 것을 방지하기 위해 사용되었다.

Wang[6]은 10 계층의 FCN 구조를 제안했다. 즉, FCN에서 모든 계층은 컨볼루션과 활성 함수의 조합으로만 구성된다. ReLU와 제로 패딩, 잔차 학습 기법이 적용되었다..

Dai[7]는 HEVC의 가변 블록 크기 (64 ~ 8)에 대응하기 위해 4 계층의 이중 경로 구조 (실제로는 6 계층)를 사용했다. 모든 계층은 컨볼루션 및 ReLU 활성함수로 구성되었으며 제로 패딩이 사용되었다. 즉, 1 계층의 출력은 2 계층과 3 계층에 동시에 전달되고 두 계층의 출력이 결합(concatenation)되어 4 계층과 5 계층으로 동시에 전달된다. 다시 두 출력은 결합되어 마지막 계층으로 전달되고 이 출력에 최초의 입력 값이 스킵 연결된다.

Meng[8]은 장기 기억(long-term memory)의 개념을 적용한 네트워크 구조를 제안했다. 저자에 따르면 제안된 모델은 FENet, UCells 및 Recon Fusion Net이라는 3가지 모듈로 구성된다. 또한 모델은 3개의 채널이 병렬적으로 구성되며 각 채널은 FENet과 UCells이 순차적으로 연결되는데 채널별로 UCells의 수가 다르게 (채널 별로 1, 2, 3개) 설정되어 있다. 3개의 채널은 Recon Fusion Net에서 하나로 합쳐져 최종적인 출력이 이루어진다. FENet은 컨볼루션 계층이며 각 채널마다 서로 다른 커널 크기를 갖는다고 언급되었으나 구체적이지 않다. 뒤이어 구성되는 UCells는 4개의 ResNet으로 구성되며 최종적인 출력과 더불어 각 ResNet의 모든 출력이 결합되어 최종적인 UCells의 출력을 이룬다. 또한, 각 ResNet의 내부 구성은 팽창된 컨볼루션과 PReLU 조합이며 3개의 서로 커널 크기를 갖는 계층이 병렬적으로 배치되어 있다. 최종적으로 Recon Fusion Net에서 모든 채널의 출력과 함께 각 UCells에서 출력된 값들이 하나로 결합되고 최초의 입력값은 스킵 연결된다.

Song[9]은 8 계층 구조를 사용했다. 마지막 계층만이 활성함수 없는 단일 컨볼루션으로 구성되고, 이전의 7 개의 계

층들은 컨볼루션-배치정규-ReLU 활성 함수의 조합으로 구성되었다.

2. 데이터셋

Park은 HM-16.0을 사용하여 HEVC에서 8 개의 테스트 시퀀스를 부호화하였다. All Intra (AI) 구성의 경우 각 시퀀스의 1, 6, 11 및 16 번째 프레임을 학습에 사용했다. Low Delay-P (LDP) 및 Random Access (RA) 구성의 경우 각 시퀀스의 0 번째, 2 번째 및 3 번째 프레임이 학습에 사용되었다. 즉, 각 시퀀스의 일부 프레임을 교육에 사용하고 해당 시퀀스의 전체 프레임에 대해 최종 테스트를 수행하여 실험 결과에 대한 비판을 받았다. 데이터셋은 양자화 파라미터 (quantization parameter: QP)에 따라 2개의 범주로 구분하였는데 범주 1에는 QP-20~29, 범주 2에는 QP-30~39로 부호화된 데이터로 구성되었다. 두 범주의 데이터셋에 대해 모델을 개별적으로 학습했으며 학습에는 Y 채널만을 사용하였지만 테스트에는 모든 YUV 채널이 적용되었다.

Wang은 BSDS500 데이터셋[10]을 HM16.0 버전으로 부호화하고 학습 (training)에 200장, 테스트 (testing)에 200장, 검증 (validation)에 100장을 사용했다. HEVC의 Common Test Condition (CTC)에 따라 QP-22, 27, 32, 37로 각각 부호화되었고 스트라이드 (stride) 21을 사용하여 38×38 크기의 패치(patch)로 분할하여 사용하였다. 또한, 데이터는 HEVC의 변환 단위 (transform unit: TU) 크기에 따라 개별적으로 수집되었다. 모델의 평가를 위한 데이터셋은 HEVC의 클래스 B에서 클래스 E까지 16 개의 테스트 시퀀스를 사용했다. 학습에는 Y 채널만 사용되었으며 테스트는 모든 채널에 대해 수행되었다.

Dai는 BSDS500 데이터셋의 400개 이미지를 사용했다. 이미지는 각각 QP-22, 27, 32, 37에서 HM-16.0의 AI 프로파일에서 후처리 필터 기능 없이 부호화하였다. 35×35 패치 크기로 겹치는 부분없이 데이터셋을 생성하였고 학습에는 Y 채널만 사용하였고 역시 테스트는 모든 채널에 대해 수행되었다. 평가를 위해 Class F를 제외한 5 개의 클래스에서 20 개의 시퀀스가 사용되었고 모든 시퀀스의 첫 번째 프레임만 사용되었다.

Meng은 570 개의 학습용 시퀀스와 582 개의 테스트용 시퀀스가 있는 Hollywood2 Scenes 데이터셋[11]을 사용하였다. 각 시퀀스에 대해 중반부 30장의 연속된 프레임 만 사용되었다. 각각의 시퀀스는 HM-7.0 버전에서 QP-22, 27, 32, 37 및 AI, LD, RA 프로파일로 부호화되었다. 각 QP별로 개별적인 네트워크를 학습하였고 학습 과정에서 더 많은 정보를 주기 위해 QP-1, QP 및 QP + 1로 부호화된 학습 데이터가 모두 사용되었다. 또한 CU (coding unit) 및 TU 경계 정보가 사용되었으나 구체적이지 않다. 이미지 패치 크기는 35×35이며 Y 채널만 학습에 사용되었다. 평가는 모든 채널에 대해 수행되었다. 클래스 B에서 F까지 총 20 개의 시퀀스에서 각각 100 개의 프레임을 사용하였다.

Song은 차세대 영상 부호화 표준화에서 사용된 JVET 참조 소프트웨어 JEM 7.0[12]을 사용하였다. Visual genome (VG)[13], DIV2K[14] 및 ILSVRC2012[15] 데이터셋을 학습에 사용하였으며 각 이미지는 필터링 옵션없이 QP-22, 27, 32, 37 및 AI 프로파일로 부호화되었다. 또한, 부호화된 이미지의 모든 픽셀에 적용된 QP 정보 맵이 추가적인 입력 값으로 사용되었다. 전체 YUV 채널에 대해 35×35 패치 크기를 사용하였고 배치 크기는 64였다. 평가용 데이터는 클래스 A1, A2 및 B~J까지의 JVET 테스트 시퀀스를 사용했다.

3. 학습 방법

표 2는 본 논문에서 분석하는 다섯 가지 CNN 기반 화질 개선 기법의 각 모델의 활성화 함수, 역전파 알고리즘, 손실 함수 및 딥러닝 프레임워크와 부호화에 사용된 HM 버전을 정리한 것이다.

표 2. 학습 조건 비교
Table 2. Comparison of Training Conditions

Author	Activation	Optimizer	Loss	Framewok	HM Ver.
Park	ReLU	SGD	MSE	Mat Conv Net	16.0
Wang	ReLU	AdaDelta	MSE	Caffe	16.0
Dai	ReLU	SGD	MSE	Caffe	16.0
Meng	PReLU	AdaDelta	L1 + MS-SSIM	-	7.0
Song	ReLU	SGD	MSE + Regularizers	Caffe	JEM 7.0

Park은 MatConvNet[16]을 사용했다. Kim[4]과 같이 높은 학습률과 gradient clipping 기술이 사용되었고, SGD (Stochastic Gradient Descent)와 MSE (Mean Squared Error)가 역전파 알고리즘과 손실 함수로 사용되었다.

Wang은 Caffe[17]를 사용하여 각 QP에 대해 모델을 개별적으로 학습하였다. QP-37 모델만이 무작위로 초기화되었고 나머지 QP 모델은 QP-37 모델로부터 전이 학습 (transfer learning)을 사용하였다. AdaDelta와 MSE는 각각 역전파와 손실 함수로 사용되었다. QP-22, 27, 32, 37 각각에 대해 1, 0.1, 0.1 및 0.01의 전역 학습률이 사용되었으며 모델의 마지막 계층 만이 전역 학습률의 1/10로 적용되었다. 또한, 학습 률의 단계적 감소 정책은 10-1 ~ 10-5 범위에서 40 epoch 마다 이루어졌다.

Dai는 Caffe, MSE 및 SGD를 사용하였다. 학습 속도를 가속화하기 위해 [-τ/α, τ/α] 범위의 gradient clipping이 사용되었으며, 여기에서 τ 는 상수로 10-2이며 α 는 기울기이다. 배치 크기는 64이며 무작위로 섞여 사용되었다. 가중치의 초기화는 He 초기화[18]기법을 사용하였고 가중치는 10-4으로 decay 되었다. 학습률은 40 epoch 마다 10-1에서 10-4 으로 decay 되었다. 바이어스 학습율은 각각 QP-27, 32 및 37에 대해 10-2,10-2 및 10-1이며 QP-22의 경우 QP-27 모델 기반의 fine tuning을 적용하여 학습률 10-3 및 바이어스 학습률 10-4 으로 40 epoch만 학습하였다.

Meng의 모델에서는 QP-37 모델의 파라미터만 무작위로 초기화되었으며 QP-22, 27 및 32는 전이 학습을 사용했다. AdaDelta 및 L1 + MS-SSIM[19] 손실 함수가 사용되었다. 각 QP 모델에 대해 학습률은 0.01, 0.1, 0.1, 1이며 바이어스 학습률은 기본 학습률의 1/10이고 gradient clipping이 적용되었다.

Song은 파라미터를 무작위로 초기화하였다. 손실 함수는 MSE와 2 개의 regularizer조합이 사용되었다. SGD가 역전파 알고리즘으로 사용되었으며 gradient clipping을 적용하였다. 학습률은 0.1, 손실 함수의 하이퍼 파라미터 λ_w, λ_s, 및 λ_{lad}는 각각 10e-5, 5e-8 및 3e-6으로 설정되었다.

Ⅴ. 비교 분석

Park 등은 빠른 모델링을 위해 Dong의 모델에 잔차 학습 기법을 결합하여 간단하게 적용하였다. 성능은 인트라 코딩과 인터 코딩에서 모두 도출되었지만, 학습과 평가에 동일한 영상 시퀀스를 사용했기 때문에 일반화 문제가 지적되었다. 여기에 적용된 잔차 학습 기법은 대부분의 화질 개선 연구에서 사용되는 것으로 깊은 네트워크를 최적화하는 데 도움이 될 뿐만 아니라 입출력 사이의 미세한 차이, 즉 노이즈를 감지하고 보완하는 측면에서 효과적이다.

Wang과 Song의 모델은 구조면에서 매우 유사하다고 볼 수 있지만, Song의 모델이 압축 표준화와 하드웨어를 더 많이 고려한 것으로 보인다. Song의 모델은 각 QP에 대해 개별적인 모델을 학습했던 이전의 연구들과 달리 모든 QP에 대해 단일 모델을 적용했다. 즉, 일반화 성능을 확보하기 위해 배치 정규화가 계층에 추가되었고 부호화된 영상의 QP 맵이 데이터셋에 추가되었다. Wang의 모델은 TU별로 데이터를 수집하여 영상 부호화 특성을 반영하였으나 Song의 모델이 보다 근본적인 문제를 해결했다고 볼 수 있다. 또한, Song의 모델은 파라미터 수를 줄이기 위해 네트워크 가지치기 (network pruning) 기법을 적용하여 약 49 %의 파라미터를 줄였다.

Dai와 Meng의 모델은 다중 경로 구조를 사용한 점에서 유사점을 찾을 수 있다. 그러나 Dai의 모델은 인트라 코딩에만 적용되었으며 저자가 언급했듯이 모든 HEVC 시퀀스에서 개선 성능을 얻지는 못했다. Meng의 모델은 다른 네트워크들에 비해 많은 수의 계층을 가지고 있지만 반대로 매개변수의 수는 상대적으로 적다. 이것은 각 층에서 추출된 특징 맵의 수를 매우 작게 설정하고 밀도가 높은 스킵 연결의 사용으로 장기 기억 메모리와 같은 기법을 적용한 덕분이다. 또한, 팽창된 컨볼루션을 사용하여 파라미터를 증가시키지 않으면서 receptive field를 확장하여 상세한 특징 맵을 입력으로부터 추출하였고 모델의 성능을 향상시켰다.

Ⅵ. 구현 및 실험 결과

모델 비교를 위한 실험은 Park, Dai 및 Kim의 모델에 대해 수행되었다. Kim의 모델의 경우 H.265/HEVC 영상의 화질 개선을 위해 적용된 것은 아니지만 네트워크 구조적 특성에 따른 학습을 결과를 비교하고자 실험 결과에 포함시켰다. 실험한 3가지 모델은 FCN 구조와 잔차 학습 기법

을 사용했다는 공통점이 있다. 반면에 네트워크의 깊이 측면에서 Park (3 계층), Dai (6계층), Kim (20계층) 순으로 차이를 보이며 네트워크 경로의 구성에서 Park과 Kim의 모델은 단일 경로, Dai의 모델은 다중 경로라는 차이가 있다. 동등 비교를 위해 각 모델은 텐서플로우[20]를 사용하여 구현되었으며 네트워크 구조는 각 논문의 설명을 따랐다. 세부적인 개발 환경의 구성은 표 3과 같다.

표 3. 구현 환경 정보
Table 3. Implementation Environment

HW	MB	ASUS Z10PE-D8
	CPU	Xeon E5-2620 v4
	RAM	128 GB
	GPU	GTX 1080 Ti
SW	Python	3.6
	Tensorflow	1.11

학습을 위한 데이터셋은 4K 화질을 갖는 RAISE 데이터셋[22]을 사용하였으며 무작위 이미지 200장을 1/4 크기로 축소한 후 HM 16.0에서 AI 및 RA 프로파일을 사용하여 QP-22, 27, 32, 37로 각각 부호화하고 QP 별로 모델의 학습을 따로 수행하였다.

부호화된 이미지는 35×35 크기의 패치로 분할하여 사용하였고 Kim의 모델에 대해서만 41×41 크기를 사용했다. Kim의 모델의 경우 3×3 커널 크기를 갖는 20 계층의 깊은 네트워크 구조를 사용하는데 전체 네트워크에 걸쳐 정상적인 특징 맵이 추출되기 위해서 요구되는 최소한의 패치 크기를 만족해야 학습에 성공할 수 있다. 배치 사이즈는 128로 설정했다. 역전파 알고리즘은 Adam[21]을 사용했으며 손실 함수는 MSE를 사용했다. Park과 Dai의 모델에 대해서는 두 가지 학습률이 적용되었는데 마지막 컨볼루션 계층에 대해서 1e-5이며 이를 제외한 나머지 계층에 대해서 1e-4이다. Kim의 모델의 경우 학습률과 가중치 decay에 1e-4가 적용되었다. 학습은 모든 모델에 대해 100만번을 수행했으며 약 1일 정도의 시간이 소요되었다.

원 저자들과는 다른 딥러닝 프레임워크와 데이터셋을 사용하였고 네트워크 구조를 제외한 세세한 파라미터들의 설정 및 역전파 알고리즘의 차이가 있음에도 불구하고 학습은 성공적으로 이루어졌다.

평가를 위한 데이터는 H.265/HEVC의 CTC (common test condition) 에 따라 HM 테스트 시퀀스 중 18개의 시퀀스를 사용하였다. HM의 AI 및 RA 프로파일을 사용하여 QP-22, 27, 32, 37로 각각 부호화하였으며 모든 영상에서 초반 10프레임만이 평가를 위해 사용되었다.

표 4는 AI 프로파일을 사용하여 QP-37로 부호화한 결과를 나타낸다. QP별로 모델 학습을 따로 시켰으나 QP가 작아질수록 PSNR 개선량은 낮아졌으며 영상 별 차이는 표 4와 유사하게 나타났다. 또한, RA 프로파일의 경우 10프레임 중 인트라-코딩만으로 부호화된 첫 번째 기준 프레임은 PSNR이 올랐지만 인터-코딩을 사용하는 나머지 프레임들은 화질 차이가 없거나 오히려 떨어졌다.

그림 1은 화질 개선량에 대한 이해를 돕기 위해 AI 프로파일을 사용하여 QP-51로 부호화 된 BasketballDrill 영상의 첫 프레임 (왼쪽)과 CNN으로 화질이 개선된 프레임 (오른쪽)을 비교한 것이다. 두 프레임 간의 화질 차이는 약 0.2dB이다. 부호화 된 프레임의 블로킹 열화가 CNN에 의해 개선되는 것을 직관적으로 확인할 수 있다. 이와 같이 PSNR 수치상에서 소수점 첫째자리의 차이는 시각적으로 인지 가능한 수준의 화질 차이를 보인다.

실험 결과 Park의 모델이 전체 시퀀스에 대해 안정적인 성능을 나타내고 있음을 알 수 있다. 이에 비해 Dai와 Kim의 모델은 일부 시퀀스에서 오히려 화질이 떨어지는 것을 볼 수 있다. 특히, Kim의 모델의 경우 다른 모델들에 비해 전반적으로 가장 낮은 성능을 보여주는 것으로 나타났다. 실험에 따르면 깊은 네트워크 구조의 이점을 찾기는 어려웠으며 오히려 가장 단순한 모델이 가장 좋은 성능을 보였다.

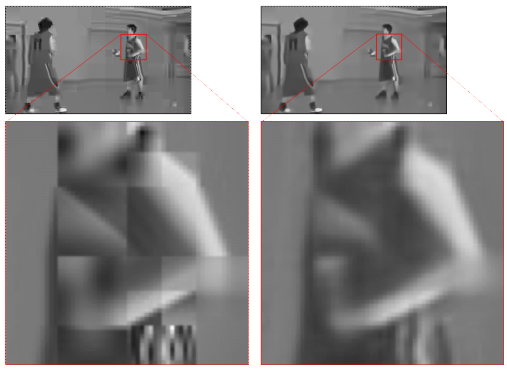

그림 1. 시각적인 화질 차이 비교
Fig. 1. Comparison of visual quality difference

표 4. HM 부호화 영상 및 CNN 기반 화질 개선 영상 간 PSNR 비교
Table 4. PSNR Comparison between compressed video using HM and improved video with CNN

Sequence	Resolution	Model	HM(QP=37)	CNN	⊿PSNR
BasketballDrill	832×480	Park	32.868	33.028	0.160
		Dai	32.868	33.046	0.179
		Kim	32.868	32.978	0.111
BasketballDrillText	832×480	Park	32.900	33.097	0.197
		Dai	32.900	33.108	0.208
		Kim	32.900	33.017	0.116
BasketballDrive	1920×1080	Park	36.171	36.215	0.044
		Dai	36.171	36.160	-0.010
		Kim	36.171	36.045	-0.126
BasketballPass	416×240	Park	32.453	32.554	0.101
		Dai	32.453	32.616	0.163
		Kim	32.453	32.445	-0.008
BlowingBubbles	416×240	Park	30.172	30.260	0.088
		Dai	30.172	30.280	0.108
		Kim	30.172	30.194	0.023
BQMall	832×480	Park	31.818	31.946	0.129
		Dai	31.818	31.925	0.107
		Kim	31.818	31.820	0.002
BQSquare	416×240	Park	29.615	29.922	0.307
		Dai	29.615	29.906	0.291
		Kim	29.615	29.841	0.225
BQTerrace	1920×1080	Park	31.507	31.562	0.056
		Dai	31.507	31.615	0.109
		Kim	31.507	31.507	0.001
Cactus	1920×1080	Park	33.101	33.172	0.071
		Dai	33.101	33.202	0.100
		Kim	33.101	33.089	-0.012
RaceHorses	832×480	Park	31.258	31.328	0.070
		Dai	31.258	31.338	0.080
		Kim	31.258	31.240	-0.018
ChinaSpeed	1024×768	Park	33.839	33.895	0.056
		Dai	33.839	33.848	0.009
		Kim	33.839	33.587	-0.252
FourPeople	1280×720	Park	35.297	35.519	0.222
		Dai	35.297	35.494	0.197
		Kim	35.297	35.412	0.114
Johnny	1280×720	Park	37.103	37.142	0.039
		Dai	37.103	37.065	-0.038
		Kim	37.103	36.971	-0.132
Kimono1	1920×1080	Park	37.348	37.359	0.011
		Dai	37.348	37.298	-0.051
		Kim	37.348	37.289	-0.060
KristenAndSara	1280×720	Park	36.669	36.863	0.194
		Dai	36.669	36.751	0.082
		Kim	36.669	36.712	0.043
ParkScene	1920×1080	Park	33.101	33.118	0.017
		Dai	33.101	33.117	0.016
		Kim	33.101	33.041	-0.060
PartyScene	832×480	Park	28.793	28.910	0.117
		Dai	28.793	28.912	0.119
		Kim	28.793	28.817	0.024
PeopleOnStreet	2560×1600	Park	33.878	34.099	0.221
		Dai	33.878	34.115	0.237
		Kim	33.878	34.003	0.125

Ⅶ. 결 론

본 논문을 통해 영상 부호화 기술을 사용하여 압축된 영상의 화질을 향상시키기 위해 CNN 기반의 접근법을 사용한 연구들을 요약하고 분석하였다. 많은 네트워크 구조가 제안되었지만 대부분 FCN이라는 구조적 유사성을 띄고 있다. 또한 영상 부호화의 다양한 특성, 즉 다양한 블록 크기와 이에 따른 화질 열화를 다루기 위해 데이터셋에 추가적인 정보를 사용하거나 병렬적인 다중 경로 구조를 채택하였다.

네트워크 구조에 따른 성능의 차이를 분석하기 위해서 제안되었던 대표적인 구조들을 직접 구현하여 성능을 비교분석하였다. 실험 결과로 보건데 화질 개선을 위해 반드시 깊은 네트워크가 최적의 성능을 보장하지는 않는다는 것을 확인할 수 있었으며 병렬적인 계층 구조 역시 부호화 영상의 특성을 반영하여 성능을 높이기에는 부족하다고 판단된다. 미래에는 다양한 모델에 대해 양질의 데이터셋을 사용하여 성능을 비교하고 분석하는 작업이 필요하다.

참 고 문 헌 (References)

[1] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, Thomas Wiegand, et al., “Overview of the high efficiency video coding (hevc) standard,” IEEE Transactions on circuits and systems for video technology, vol.22, no. 12, pp. 1649 – 1668, 2012.

[2] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang, “Image super-resolution using deep convolutional networks,” IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 2, pp. 295 – 307, 2016.

[3] Ke Yu, Chao Dong, Chen Change Loy, and Xiaoou Tang, “Deep convolution networks for compression artifacts reduction,” arXiv preprint arXiv:1608.02778,2016.

[4] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee, “Accurate image super-resolution using very deep convolutional networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp.1646 – 1654.

[5] Woon-Sung Park and Munchurl Kim, “Cnn-based in-loop filtering for coding efficiency improvement,” in Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), 2016 IEEE 12th. IEEE, 2016, pp.1 – 5.

[6] Tingting Wang, Mingjin Chen, and Hongyang Chao, “A novel deep learning-based method of improving coding efficiency from the decoder-end for hevc,” in Data Compression Conference (DCC), 2017. IEEE, 2017, pp.410 – 419.

[7] Yuanying Dai, Dong Liu, and Feng Wu, “A convolutional neural network approach for post-processing in hevc intra coding,” in International Conference on Multimedia Modeling. Springer, 2017, pp. 28 – 39.

[8] Xiandong Meng, Chen Chen, Shuyuan Zhu, and Bing Zeng, “A new hevc in-loop filter based on multi-channel long-short-term dependency residual networks,” in 2018 Data Compression Conference. IEEE, 2018, pp. 187 – 196.

[9] Xiaodan Song, Jiabao Yao, Lulu Zhou, Li Wang, Xiaoyang Wu, Di Xie, and Shiliang Pu, “A practical convolutional neural network as loop filter for intra frame,” arXiv preprint arXiv:1805.06121, 2018.

[10] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik, “Contour detection and hierarchical image segmentation,” IEEE transactions on pattern analysis and machine intelligence, vol. 33, no. 5, pp. 898 – 916, 2011.

[11] Marcin Marszalek, Ivan Laptev, and Cordelia Schmid, “Actions in context,” in Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009, pp. 2929 – 2936.

[12] JEM 7.0, https://jvet.hhi.fraunhofer.de/trac/vvc/browser/jem/branches/HM-16.6-JEM-7.0-dev, 2019, [Online; accessed February 3, 2019].

[13] Visual genume (VG), http://visualgenome.org/, 2019, [Online; accessed February 3, 2019].

[14] DIV2K, https://data.vision.ee.ethz.ch/cvl/DIV2K/, 2019, [Online; accessed February 3, 2019].

[15] ILSVRC2012, http://www.image-net.org/challenges/LSVRC/2012/, 2019, [Online; accessed February 3, 2019].

[16] Andrea Vedaldi and Karel Lenc, “Matconvnet: Convolutional neural networks for matlab,” in Proceedings of the 23rd ACM international conference on Multimedia. ACM, 2015, pp. 689 – 692.

[17] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014, pp. 675 – 678.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1026 – 1034.

[19] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz, “Loss functions for image restoration with neural networks,” IEEE Transactions on Computational Imaging, vol. 3, no. 1, pp. 47 – 57, 2017.

[20] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al., “Tensorflow: A system for large-scale machine learning,” in 12th{USENIX} Symposium on Operating Systems Design and Implementation ({OSDI}16), 2016, pp. 265 – 283.

[21] Diederik P Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv: 1412. 6980, 2014.

[22] Duc-Tien Dang-Nguyen, Cecilia Pasquini, Valentina Conotter, and Giulia Boato, “Raise: A raw images dataset for digital image forensics,” in Proceedings of the 6th ACM Multimedia Systems Conference. ACM, 2015, pp. 219 – 224.

저 자 소 개

이 영 운

- 2016년 : 선문대학교 행정학과 행정학사
- 2016년 : 선문대학교 컴퓨터공학과 공학사
- 2018년 : 선문대학교 컴퓨터융합전자공학과 석사
- 2018년 ~ 현재 : 선문대학교 컴퓨터융합전자공학과 박사과정
- ORCID : https://orcid.org/0000-0001-9011-0921
- 주관심분야 : 디지털영상처리, 압축영상화질개선, 기계학습

김 병 규

- 1998년 한국과학기술원 전기및전자공학과 석사
- 2004년 한국과학기술원 전기및전자공학과 박사
- 2004년 ~ 2008년 : 한국전자통신연구원 선임연구원
- 2009년 ~ 2015년 : 선문대학교 컴퓨터공학과 부교수
- 2016년 ~ 2018년 : 숙명여자대학교 IT공학과 부교수
- 2019년 ~ 현재 : 숙명여자대학교 IT공학과 교수
- ORCID : https://orcid.org/0000-0001-6555-3464
- 주관심분야 : 영상 및 비디오 신호처리, 패턴 인식, 딥 러닝 기반 시각 지능 알고리즘, 비디오 압축 표준기술