블록 기반 특징맵 크기 조정을 이용한 DNN 특징맵 압축

윤규리, 정혜원, 김영웅, *김연희, *정세윤, 김휘용†
경희대학교, *한국전자통신연구원
{curieyoon, woni980911, duddnd7575, hykim.v} @ khu.ac.kr,
* {kimyounhee, jsy} @ etri.re.kr

Neural Feature Compression with Block-based Feature Resizing

Curie Yoon, Hye Won Jeong, Yeongwoong Kim, * Younhee Kim,
* Se-Yoon Jeong and Hui Yong Kim
Kyung Hee Univiersity, * Electronics and Telecommunications Research Institute

요 약

자율주행, IoT 등 많은 양의 영상 정보를 실시간으로 처리해야 하는 기술과 mobile device 등의 기기에서 Machine Learning 연산을 하는 소프트웨어들이 등장함에 따라 사람을 위한 영상을 출력하는 영상 부호화 기술 대신 기계의 vision task 성능을 위해 특화된 영상 부호화 기술의 필요성이 대두됐다. 본 연구에서는 영상에서 추출한 특징맵을 Neural-Net based Video Coding 모델을 이용해 압축률과 기계의 vision task 성능을 동시에 최적화한다. 또한, 하드웨어 친화적인 block-based 처리와 이로 인한 성능 저하를 최소화하기 위해 적응적 resizing 방식을 제안한다.

1. Introduction

1.1 연구배경

최근 자율주행자동차, 사물인터넷(IoT), 스마트 시티 등 여러 응용 분야에서는 영상 및 이미지 데이터를 사용한 딥러닝 기반 머신 비전 기술이 이용되고 있다. 머신 비전 기술은 입력된 영상(이미지)로부터 머신 비전을 수행하기 좋은 표현인 특징맵을 추출한 후 이를 분석해 객체 탐지, 객체 분류 등의 특화된 vision task 를 수행한다. VCM(Video Coding for Machines)은 머신 비전을 수행하기 위해 입력 영상을 압축하는 대신 영상으로부터 추출한 특징맵을 압축한다.

특징맵 압축은 기존의 전통적인 영상 압축 방식인 HEVC, VVC 또는 깊은 신경망 기반의 모델 등 기존의 HVS(Human Vison System) 기반 영상 압축 방식을 통해 이루어질 수 있다. 두 방식 모두 Human Vision 을 위해 고안된 모델이므로 압축률의 한계(400 배)라는 bottleneck 이 존재해 오늘날 기계가 처리할 방대한 양의 영상을 압축하기에는 매우 부족하며, vision task 를 수행하기 위해 원본 영상을 통째로 복원할 필요성은 적다. [1] 또한, 기존의 Video Coding 이 쓰이는 환경에서는 모델의 복잡도나 연산량을 고려할 필요가 적었으나, 기계를 위한 Video Coding 이 쓰이는 환경에서는 자율주행, IoT 등 vision task 를 실시간으로 처리해야 하는 경우가 많기에 모델의 복잡도와 연산량도 고려해야 한다. 또한, Neural Image Compressor 를 특징맵 압축에 사용하기 위해서는 모델을 처음부터 다시 학습시켜야 하며, 입출력의 채널 수와 형식 등이 다르므로 이에 대응해야 한다.

HVS 기반의 비디오 코딩 방식 대신 VCM 을 사용할 경우 특징맵 추출과 특징맵 분석이 분리돼 서로 다른 하드웨어에서 이루어지는 load balancing 이 가능해진다. 이는 특히 최근 단말의 하드웨어 성능 발달로 여러 응용분야에서 이용될 수 있다. 또한 일반 영상 압축 방식을 사용할 경우 전송한 데이터의 압축을 풀면 그대로 원본 영상이 복원돼 정보 보안의 문제가 발생할 수 있으나, 특징맵은 악의적인 노드에 의해 복원되더라도 기계만이 인식할 수 있는 특징맵이 복원되기 때문에 프라이버시 문제에서도 자유롭다. [2] 따라서 영상의 특징맵을 압축하는 저복잡도의 기계를 위한 영상 부호화 기술(Video Coding for

Machines)의 필요성이 더욱 두드러진다.

1.2 연구목표

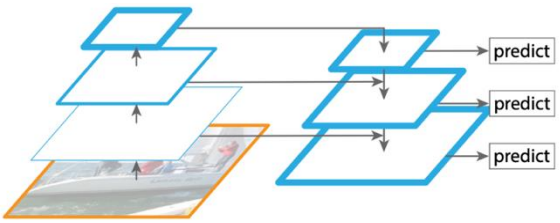
VCM 은 특징맵 압축 효율과 기계의 vision task 성능을 공동최적화(joint optimization)하는 것을 목표로 한다. vision task 를 수행하기 전 영상을 압축해 송수신하는 대신 영상을 압축하기 전 영상에서 추출한 특징맵을 압축해 압축 효율과 vision task 성능을 높이고자 한다. 특징맵 압축 전 원본 영상에서 Feature Pyramid Network(FPN)을 이용해 특징맵을 추출하고, 복원된 특징맵은 Faster R-CNN 으로 객체 탐지(Object Detection)를 수행해 성능을 측정한다.

영상 압축 단계에서 8 개의 양자화 레벨(Quantization Step)에 따라 가장 적은 비트스트림과 낮은 특징맵 복원 수준부터 가장 큰 비트스트림과 높은 특징맵 복원 수준까지 VCM 의 성능을 측정해, 낮은 비트스트림과 높은 비트스트림에서 고루 준수한 성능을 낼 수 있게 한다.

본 연구에서는 더 나아가 하드웨어에서의 메모리 이슈를 고려한 Block-Based 모델을 제안해 큰 스케일에서 하드웨어 친화적인 방식으로 모델을 사용할 수 있으며 Block-Based 처리를 하지 않은 모델로부터 성능 저하를 방지했다.

2. Related Works

2.1 Feature Extraction 과 Feature Analysis(object detection)



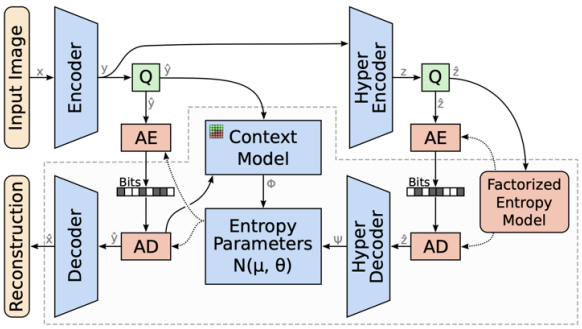
[그림 1]

영상의 객체 추적은 영상 내의 여러 크기의 객체들을 탐지하기 위해 특징맵을 여러 스케일에서 추출한다. 특징맵 추출 과정은 높은 해상도의 저차원 특징맵(high-resolution, low-level features)부터 낮은 해상도의 고차원 특징맵(low-resolution, high-level features)까지 bottom-up 방식으로 5 개의 특징맵을 추출한다. 추출한 특징맵들에 convolution 을 거쳐 가장 고차원 특징맵부터 자신 직전에 추출된 특징맵에 크기를 맞춘 후 element-wise 로 더해준다. [그림 1] 은 좌측에서 bottom-up 방식의 CNN 연산을 거친 후 우측에서 top-down 방식으로 각 크기의 특징맵에 상위 특징맵들을 더하는 과정을 나타낸다. 이 방식으로 두 번째로 저차원 특징을 가진 특징맵까지 모든 레벨의 특징맵의 정보를 가지게 된다. FPN 은 최종적으로 얻은 5 개의 특징맵들에 한 번 더 convolution 연산을 거쳐 5 개 해상도의 각각 다른 특징맵들을 출력한다. [3] 본 연구에서는 출력된 FPN 특징맵을 사용해 압축과 복원을 거친 후 객체 탐지에 사용했다.

Faster R-CNN 은 영상으로부터 추출한 특징맵에 대해 Region Proposal Network(RPN)을 사용해 경계 상자에 대한 예측(bounding box proposal)과 각 class score 를 출력한다. FPN 을 사용한 Faster R-CNN 은 FPN 으로부터 여러 크기의 특징맵을 받으므로 각 특징맵에 대해 1 개의 anchor box 를 이용해 경계 상자 예측을 한다. 특징맵들은 RPN 과 RoI(Region of Interest) Pooling 을 거친 후 convolution 을 통해 각각 분류와 경계 상자 탐지를 위한 2 개의 branch 로 나뉘어 결과로 각 객체의 bounding box 와 class 를 출력한다. [4]

2.3 Neural-Network based Video Coding (NNVC)

Neural-Network based Video Coding 은 깊은 신경망 학습을 통해 영상의 공간적 중복성(spatial redundancy)을 제거해 전통적인 비디오 코딩 방식에 비해 압축 효율과 영상 복원 품질을 향상시켰다. [5] 입력 이미지 x 는 분해변환(analysis transform)을 거쳐 잠재표현(latent representation) y 로 변환된다. y 에 한 번 더 분해변환을 거쳐 초사전정보(hyper-prior) z 를 얻는다. 양자화된 $\hat{z}$ 는 산술부호화 과정을 거쳐 비트스트림으로 전송되며, 복원된 $\hat{z}$ 는 합성변환(synthesis transform)을 통해 $\hat{y}$ 의 산술부호화에 쓰일 엔트로피 모델의 가우시안 평균 μ 와 표준편차 θ 를 출력한다. 복원된 $\hat{y}$ 는 합성변환을 통해 최종적으로 복원된 이미지 $\hat{x}$ 를 출력한다. [6]에서는 각 이미지에서 이미 디코딩 된 잠재표현 $\hat{y}_{k<i}$ 들에 Context Prediction 연산을 적용해 복원된 $\hat{z}$와 함께 $\hat{y}_i$ 의 엔트로피 모델을 추정하는데에 사용했다. 새로운 이미지를 디코딩 할 때마다 디코딩 된 잠재표현의 값은 0 으로 초기화된다.



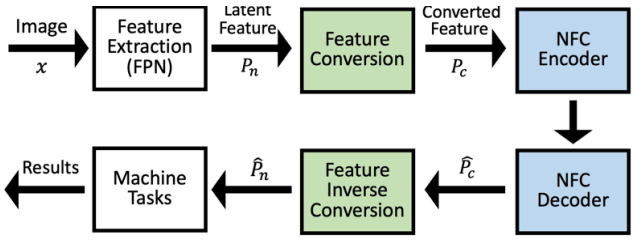
[그림 3]

[그림 3]은 NNVC 의 모델 구조이다. NNVC 는 잠재표현 $\hat{y}$와 초사전정보 $\hat{z}$를 출력하고 복원하기 위한 분해변환, 합성변환과 Context Prediction 에 모두 convolution 층으로 이루어진 깊은 신경망을 사용했다. 또, 잠재표현 $\hat{y}$ 를 연산하는 분해변환과 $\hat{y}$로부터 최종 이미지를 복원하는 합성변환에서 합성함수로 각각 GDN(Generalized Divisive Normalization)과 IGDN(Inversed GDN)을 사용해 원본 이미지의 공간적 중복성을 제거했다.

3. Model Architecture

3.1 Neural Feature map Compressor (NFC)

[그림 5]는 NFC 의 전체 pipeline 이다. 입력 이미지는 FPN 을 거쳐 각각 m^2 개 채널을 가지는 5 가지 크기의 특징맵 P_n 으로 출력된다. 특징맵들은 1 개 채널로 각각 tiling 돼 grayscale 이미지와 유사한 형태로 NFC 에 입력된다. 변환된 특징맵은 분해변환을 거쳐 압축에 유리한 잠재표현 y 로 변환된다. y 에 깊은 분해변환을 거쳐 엔트로피 모델의 평균과 표준편차를 얻기 위한 메타정보 z 를 얻은 후, 양자화된 $\hat{z}$ 는 고정된 확률분포에 따라 산술부호화 돼 비트스트림으로 압축된다. 깊은 합성변환을 거친 $\hat{z}$를 $\hat{y}$의 산술부호화 엔트로피 모델의 평균과 표준편차로 사용하고, 비트스트림으로 압축된 $\hat{z}$ 는 $\hat{y}$ 의 비트스트림과 함께 전송돼 송신 하드웨어에서 $\hat{y}$의 디코딩을 할 때 사용된다. 복원된 $\hat{y}$는 합성변환을 통해 grayscale 이미지와 유사한 1 차원의 특징맵 $\hat{x}$으로 복원된다. 이 특징맵을 m^2 개의 채널로 재배열해 최종 복원된 특징맵을 얻는다. 본 연구에서는 [6]과 달리 Context Prediction 은 사용하지 않았다.



[그림 5]

p2, p3, p4, p5, p6 특징맵은 각각 NFC 의 압축과 복원을 거친 후 객체 탐지를 위한 네트워크인 Faster R-CNN 의 특징맵 분석 부분으로 들어가 성능이 측정된다.

3.2 Block-based NFC

본 연구에서는 블록 처리에 의한 NFC 성능 저하를 방지하기 위해 2n*2n 블록별 적응적 resizing 방식을 사용했다. 특징맵 블록들 중 일부 블록은 다른 블록들에 비해 더 중요한 특징을 담고 있을 것이라는 가정에 기반해, threshold 보다 많은 정보를 담고 있는 블록들은 non-resizing 방식(mode 1)으로, 그렇지 않은 블록들은 resizing 방식(mode 2))으로 NFC 에 입력한다. Mode 1 에 해당하는 블록은 n*n 크기의 4 개의 작은 블록들로 나뉘어 NFC 에 입력된 후 복원된 작은 블록들을 재배열해 원 블록으로 복원된다. Mode 2 에 해당하는 블록은 n*n 블록으로 downsampling 후 NFC 와 upsampling 을 통해 2n*2n 블록으로 복원된다. NFC 내부에서는 각 블록의 Mode 정보를 전달해 디코딩 후 복원된 n*n 블록에 대해 upsampling 을 할지 rearranging 을 할지 결정한다. downsampling 과 upsampling 은 bicubic interpolation 을 이용한다.

mode decision 은 각 블록별로 Mode 1 과 Mode 2 의 cost 를 비교해 결정된다. cost 는 $J = \lambda \cdot MSE \cdot 255^2 + BPP$ 로 정의한다. quality 에 따른 lambda 값으로 MSE 에 가중치를 곱해 lambda 값이 높아질수록 bpp 와 품질이 더 높은 mode 1 블록이 많아진다. 각 모델과 특징맵의 스케일별로 서로 다른 lambda 를 사용했으며, lambda 별 8 단계의 품질 설정에 따라 mode 1 과 mode 2 의 배정에 따른 압축률과 품질을 테스트했다.

4. Experiments

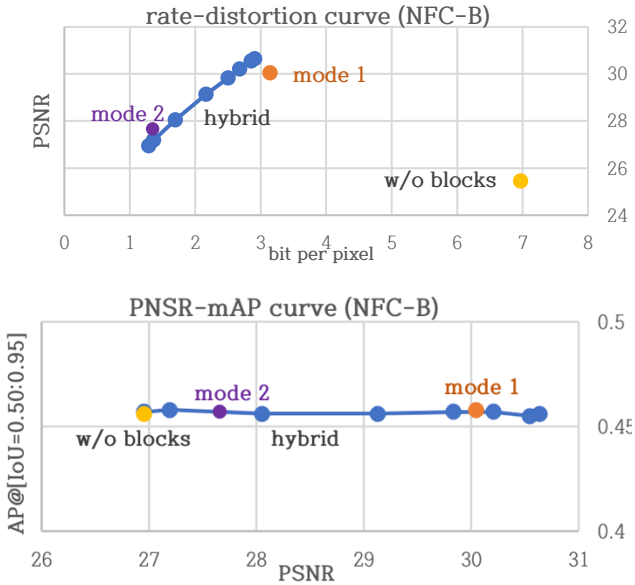
실험은 compressai 라이브러리를 이용해 내부 채널 갯수에 따라 총 가중치의 갯수가 17,542,497 개인 모델 NFC-B 를 학습시켰으며, 추가적으로 가중치의 갯수가 7,015,201 개인 NFC-A 로도 실험을 진행했다. 각 모델은 블록 처리를 하지 않는 모델과 p2, p3, p4 특징맵에 대해 mode 1 로 블록 처리를 하는 모델을 별개로 학습했으며 mode 1 로 학습한 모델을 mode 1, mode 2, hybrid(mode decision) 테스트에 사용했다. batch size 는 1, learning rate=0.0001 로 NFC-A 는 70 만회, NFC-B 는 20 만회의 가중치 업데이트 후 성능을 테스트했다.

학습과 테스트 데이터인 특징맵 추출과 복원된 특징맵의 성능 평가에는 pre-trained 모델인 facebook detectron2 의 Faster R-CNN X101 FPN 과 COCO 2017 의 subset 을 사용했다. FPN 으로 m^2 =256 개 채널의 특징맵 p2, p3, p4, p5, p6 을 추출 후 training, evaluation, test 데이터셋을 구성했다. 특징맵들은 tiling 을 통해 1 개 채널로 펼친 후 사용했으며 복원된 특징맵은 객체 탐지에 사용되기 전 다시 256 차원으로 rearranging 해 객체 탐지 성능을 측정했다. 서로 다른 모델들의 성능은 객체 탐지의 Average Precision@[IoU=0.50:0.95]로 비교했다. Uncompressed 특징맵의 AP 는 0.457 이다.

학습은 Neural-based Image Compressor 과 유사하게 입력 블록에 CNN 연산을 수행하는 가중치들과 산술부호화의 엔트로피 모델의 평균값과 표준편차를 계산하기 위한 가중치들을 최적화하는 방식으로, $J = \lambda \cdot MSE \cdot 255^2 + BPP$ 를 최소화한다. MSE 는 입력 특징맵과 복원 특징맵 간의 mean squared error 이며 bpp 는 압축된 특징맵의 비트스트림의 길이이다.

블록 처리를 하는 모델의 경우 p2, p3, p4 스케일의 특징맵에 대해서만 블록 처리를 했으며 p5, p6 은 블록 처리를 하지 않은 NFC 와 동일하게 처리했다.

4.2 Block-based NFC



[그래프 1], [그래프 2]

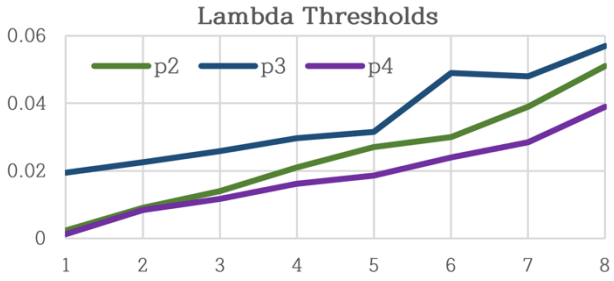
[그래프 1]은 블록 처리를 하지 않은 NFC-B 와 블록 처리를 한 NFC-B 의 세 가지 실험 결과로 나타낸 rate-distortion curve 이다. mode decision 을 사용했을 때 낮은 양자화 레벨에 대해서 mode 1 보다 개선된 성능을 냈고, 높은 양자화 레벨에서는 mode 2 를 단독 사용했을 때보다 근소하게 떨어진 성능을 냈다. 세 가지 블록 모델 모두 블록 처리를 하지 않은 NFC-B 보다 월등히 우수한 성능을 냈다.

[그래프 2]는 [그래프 1]의 PSNR 에 따른 객체 탐지의 AP 성능이다. rate-distortion curve 에서 네가지 모델이 25.449 부터 30.644 까지의 넓은 PSNR 스펙트럼을 보였으나 실제 객체 탐지 성능에는 차이가 거의 없었다. 이를 통해 원본 특징맵에 대한 PSNR 값의 차이로 실제 객체 탐지 성능의 차이를 예측하기는 어려움을 알 수 있으며, 따라서 rate-AP curve 와 rate-PSNR curve 는 거의 차이가 없음을 알 수 있다.

또한, 블록 처리를 통해 객체 탐지 성능은 유지하면서 압축률을 월등히 높일 수 있음을 알 수 있다. 실제 블록 처리를 하지 않은 모델과 mode 2 또는 hybrid-quality 1 는 약 5 배의 압축률 차이를 보였다.

4.3. mode decision threshold

hybrid NFC 는 mode 1 과 mode2 로 128*128 블록을 복원할 때 각 블록의 MSE 와 bpp 으로 각 mode 의 cost 를 계산한다. cost 는 $J = \lambda \cdot MSE \cdot 255^2 + BPP$ 으로, lambda 값이 커질 수록 MSE 의 중요도가 커져 mode 1 로 배정되는 블록이 많아진다.



[그래프 3]

특징맵 p2, p3, p4 에서 동일한 lambda 값을 사용해 실험을 한 결과, 한 개 이상의 스케일에서 모든 품질 설정에 대해 한 가지의 mode 배정만 일어나거나 스케일에 따라 mode 배정이 불균형하게 이루어졌다. 예를 들어, p2 특징맵이 품질 설정에 따라 블록의 mode 배정이 달라지게 하는 적절한 lambda 값들을 p3 특징맵에 사용할 경우 모든 품질 설정에서 모든 블록이 mode 2 로 배정되고, p4 특징맵에 사용할 경우 모든 품질 설정에서 모든 블록이 mode 1 로 배정됐다.

따라서 품질 설정에 따라 블록의 배정 결과를 다르게 하기 위해 모델의 가중치 갯수와 특징맵의 스케일 별로 별개의 lambda 값들을 설정해 품질 설정에 따라 모든 스케일의 특징맵에서 블록 배정이 다르게 이루어질 수 있도록 했다. [그래프 3]은 NFC-B 에서 특징맵의 스케일에 따라 블록들의 mode 배정이 품질 설정에 따라 다르게 이루어질 수 있도록 하는 lambda 값들을 나타낸 것이다. lambda 의 범위가 큰 특징맵 스케일일 수록 mode 1 로 블록이 배정되기 위해 큰 가중치가 필요하므로 해당 스케일에서 mode 2 처리를 하는 것이 대체로 유리함을 알 수 있다. 따라서 블록 단위의 mode decision 이 어려운 환경에서는 p3 는 모두 mode 2 로, p4 는 모두 mode 1 로 처리하고 p2 에 대해서만 mode decision 또는 단일 mode 처리를 하면 유사한 성능을 낼 수 있을 것이라 추정한다.

같은 특징맵 스케일 안에서도 특징맵에 따라 블록 배정의 편차가 큰 것을 확인할 수 있었다. 따라서 각 특징맵 블록들의 mode 1 과 mode 2 의 cost 를 비교해 특징맵 별 information density 를 측정할 수 있음을 알 수 있다.

5. Conclusion

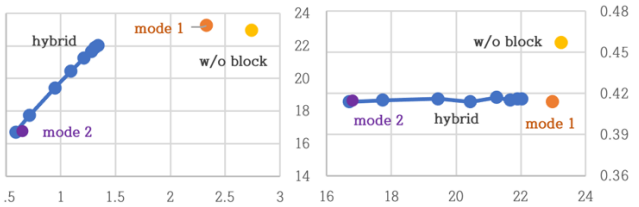
본 연구에서는 기존에 영상 압축에 사용되던 신경망 모델인 NNVC 를 특징맵 압축에도 사용함으로써 이미지 압축에 사용된 깊은 신경망 모델이 특징맵 압축에도 쓰일 수 있음을 보여준다. 특히, Block-Based 방식을 사용해 tiling 후 높은 해상도의 특징맵에 대해서도 하드웨어 친화적인 연산을 할 수 있을 뿐만 아니라, 충분히 가중치 갯수가 큰 모델의 경우 vision task 성능은 유지하면서 bpp 를 현저히 감소시켜 압축률을 향상시킬 수 있음을 보였다. 또한, 특징맵 스케일에 따른 적절한 lambda 값들을 찾음으로써 더 세밀한 품질 설정을 가능하게 하였다.

향후 원본 특징맵에 대한 MSE 가 아닌 vision task 의 성능을 통한 학습으로 모델의 성능 향상을 기대할 수 있다. mode 2 의 upsampling 에 super resolution 등의 기법을 적용해 추가적인 성능 향상을 기대할 수 있다. 마지막으로, 블록 처리시 연산 속도가 더 빠른 방식의 NFC 를 구현해 효율을 높이고자 한다.

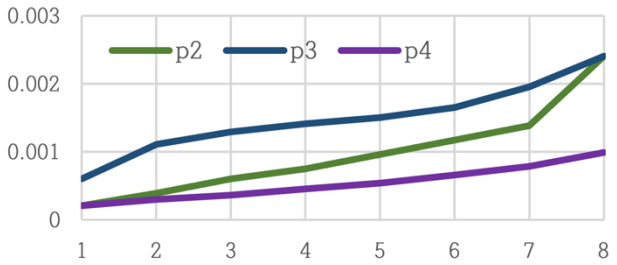
6. Appendix

[그래프 4]은 block 처리를 하지 않은 NFC-A 와 블록 처리를 한 3 가지 방식의 NFC-A 의 rate-distortion curve 이다. mode decision 을 사용했을 때 높은 양자화 레벨에서 mode 2 를 단독 사용했을 때보다 근소하게 개선된 성능을 냈다. mode 1 모델보다는 bpp 와 PSNR 값이 모두 다 낮았고, 블록 처리를 한 세 모델 모두 블록 처리를 하지 않은 모델보다 성능이 우수했다.

[그래프 5]는 [그래프 4]의 PSNR 에 따른 객체 탐지의 mAP 성능이다. NFC-B 와 달리, NFC-A 는 블록 처리를 한 모델의 객체 탐지 성능이 떨어졌으며, 모든 PSNR 값은 달랐으나 객체 탐지 성능은 거의 차이가 없었다. 특히 mode 1 은 PSNR 값은 블록 처리를 하지 않은 모델과 비슷했으나 객체 탐지 성능은 비슷해, NFC-B 와 같이 PSNR 값의 차이로 객체 탐지 성능의 차이를 예측하기 어려움을 알 수 있다.



[그래프 4], [그래프 5]



[그래프 6]

[그래프 6]은 NFC-A 에서 mode decision 을 양자화 레벨마다 다르게 하기 위한 p2, p3, p4 의 lambda 값이다. 가중치의 갯수가 많은 모델 NFC-B 에서는 NFC-A 의 lambda 보다 약 20 배 큰 값을 사용해야 mode decision 이 가능했다. 즉, 가중치의 갯수가 많은 모델에서 가중치의 갯수가 적은 모델보다 downsampling 으로 bpp 를 줄이는 것이 더 유리한 경향이 있음을 알 수 있다.

Acknowledgement

이 논문은 2021 년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2020-0-00011, (전문연구실)기계를 위한 영상부호화 기술)

참 고 문 헌

[1] ITU, “H.266:Versatile video Coding.” (2021)

[2] ISO/IEC JCT1/SC29/WG2, “Use Cases and Requirements”, wg2n00190, April, 2022.

[3] Lin, Tsung-Yi, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, “Feature Pyramid Network for Object Detection.” CVPR. (2017)

[4] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." IEEE. (2017)

[5] Ballé, Johannes, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston, “Variational Image Compression with a Scale Hyperprior.” (2018).

[6] Minnen, David, Johannes Ballé, and George Toderici. “Joint Autoregressive and Hierachical Priors for Learned Image Compression.” Arxiv: 1809.02736 (2018).