



경희대학교
KYUNG HEE UNIVERSITY

Capstone Design 2

Improvement of Image Coding Performance using Neural Network

2022.05.10

박민정



Visual Media Lab

<http://vmlab.khu.ac.kr>

Contents

1. 진행 상황
2. 질문 사항
3. 향후 일정

진행 상황

진행 상황

◆ 코드 리뷰 및 개선 모델 구조 구상

- 코드 리뷰 완료
- Step마다 모듈을 만들어서 상속 받는 형태로 구상
- 현재 downscaling은 복잡하지 않으므로 영웅 님 코드에서 추가했었지만..!

◆ Scaling 구현 및 실험

- Pre) Downscaling
 - ★ cv2.INTER_CUBIC
 - Bicubic interpolation over a 4x4 pixel neigh.
- Post) Downscaling
 - ★ Averaging filter
 - + Nearest neighbor interpolation
- Post) Upscaling
 - ★ OpenCV SR method

2022. 5						
SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3 ✓meeting	4	5	6	7
8	9	10	11	12	13 논문 제출	14
15	16	17 ✓meeting	18	19	20	21
22	23	24	25	26	27	28
29	30 최종 논문	31 ✓meeting				

- 코드 리뷰 및 개선 모델 구조 구상 (quality 0,3,5,7)
- Scaling 구현 및 실험
- 이미지 블록 기반 처리
- Mode 1 구현 및 실험
- Mode 2 구현 및 실험
- Sampling 구현 및 실험 및 논문 작성

◆ 방미공 논문

- 좀 더 구체적으로 제목 변경
- 요약 및 서론 작성

2022 년 한국방송·미디어공학회 하계학술대회

스케일링과 모드 결정법을 이용한 신경망 기반 이미지 부호화 성능 개선

저자일, *저자이
경희대학교

author1@gmail.com, *author2@naver.com

Improvement of the Neural Network based Image Coding Performance
using Scaling and Mode Decision Method

Anony Kim *Mous Lee
Kyung Hee University

요 약

본 논문에서는 최근 연구되고 있는 신경망 기반 이미지 부호화 기법(Neural Network based Image Coding, NNIC)의 성능을 개선하기 위하여 스케일링과 모드 결정법을 활용하는 방안을 제시한다. 업 스케일링 기법(Upscaling Method)과 다운 스케일링 기법(Downscaling Method)을 활용하여 낮은 비트율(bitrate)에서 좋은 성능을 얻는다. 또한 이미지를 여러 개의 $2N \times 2N$ 블록으로 분할하고, 하나의 $2N \times 2N$ 블록에 대한 4 개의 $N \times N$ 블록이 신경망 기반 이미지 부호화 모델의 입력이 되는 모드 1 과 $2N \times 2N$ 을 다운 스케일링기법을 활용하여 $N \times N$ 블록이 신경망 기반 이미지 부호화 모델의 입력이 되는 모드 2 를 활용한 모드 결정법(Mode Decision Method)을 제안한다. 모드 결정법은 비트율-왜곡 최적화 기법(Rate-distortion optimization method, RDO)이 적용된다. 인코더(encoder)에서는 모드를 저장하고, 복호화 시 사용할 수 있도록 디코더(decoder)에 전달한다. 이 방법을 활용하여 모든 비트율에서 Anchor 에 비해 성능 향상을 얻을 수 있도록 한다.

1. 서론

정보통신기술과 하드웨어의 급격한 발전으로 인해 최근에는 연구소 및 기업 뿐만 아니라 일반 사용자들도 대용량 이미지 데이터를 다루고 있다. 초고화질 이미지에 대한 수요가 증가함에 따라 이미지 부호화 기술은 중요한 위치를 차지하고 있다. 또한 GPU 의 획기적인 성능 향상으로 근래 딥러닝 분야가 큰 발전을 이루며 다양한 딥러닝 모델이 발표되고 있다. 이러한 딥러닝 모델은 여러 분야에 적용되고 있는데, 이미지 부호화 기술에도 이를 적용한 연구가 활발히 진행되고 있다. 따라서 본 연구에서는 신경망 기반 이미지 부호화 모델의 성능을 개선하는 방법으로 스케일링 기법과 비트율-왜곡 최적화 기법이 적용된 모드 결정법을 제안한다.

질문 사항

- ◆ 방미공 논문에 저자는 누구누구 써야 하나요?
- ◆ HEVC는 CTU로 나누고 YCbCr별 CTB도 나누고, CTU에서 CU 분할하고... 복잡한 걸로 기억하는데요!
저희는 그런 거 상관 없이 이미지를 픽셀 기반으로 $2N \times 2N$ 으로 나누면 되는 건가요?
만약 그렇다면 몇 픽셀 정도로 블록을 만들면 될까요?
 - 그리고 블록으로 만드는 이유가 예측을 위해서 인건가요?
- ◆ Upscaling을 opencv에서 제공하는 deep learning super resolution 메소드를 써도 될까요?
- ◆ 교수님께서 말씀하신 scaling이 단순히 resizing만 하는 게 아니고,
blurring & sharpening도 포함되어 있는 게 맞을까요?
- ◆ 모드 결정법을 통해 높은 bitrate에서만 성능 향상을 예상하시는건지 (즉, scaling+모드 결정법)
아니면 모드 결정법 자체가 모든 bitrate에서 성능 향상을 야기하는건지 (모드 결정법만 써도 성능↑)
궁금합니다!

향후 일정

2022. 5

SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3 ✓meeting	4	5	6	7
8	9	10	11	12	13 논문 제출	14
15	16	17 ✓meeting	18	19	20	21
22	23	24	25	26	27	28
29	30 최종 논문	31 ✓meeting				

- 코드 리뷰
및 개선 모델 구조 구상
(quality 0,3,5,7)
- Scaling 구현 및 실험
- 이미지 블록 기반 처리
- Mode 1 구현 및 실험
- Mode 2 구현 및 실험
- Sampling 구현 및 실험
및 논문 작성

Appendix

✓ Approach

Approach – Step 1

Step 1

Image를 input 라고 하자. NNIC는 Minnen, Lee, Cheng이 제안한 Mean Scale Hyperprior 모델을 기본으로 약간 수정된 모델이다.

VCM의 경우, input을 feature로 받으므로 input X 는 feature라고 하자. VCM에서의 블록도는 아래와 같다.



Feature Extract 과정을 통해 Image에서 Feature을 추출하고, 추출된 Feature X 를 NNIC의 입력으로 넣는다. 압축 후 복원된 Feature \hat{X} 을 Analysis Net에 넣어 Detection, Segmentation 등을 수행한다. 성능평가는 [MAP](#)을 통해 진행한다.

이미지를 입력으로 받는 NNIC의 경우, input X 는 image라고 하자. 이미지를 입력으로 받는 NNIC의 블록도는 다음과 같다.



Image X 를 NNIC의 입력으로 넣는다. NNIC 과정을 통해 압축 후 복원된 이미지 \hat{X} 를 얻는다. X 와 \hat{X} 를 통해 PSNR을 구한 후, PSNR을 통해 성능평가를 진행한다. (2022. 3. 현재 하고 있는 NNVC 실습이 이에 해당한다.)

- Base code Option
 - [facebookresearch/NeuralCompression](https://facebookresearch.github.io/NeuralCompression/)
 - [MeanScaleHyperprior](#)

Step1을 통해 얻은 결과는 **Anchor**가 되어 이후 진행할 연구의 기초 비교대상이 된다.

본 연구에서는 이미지를 입력으로 받는 NNIC를 연구하므로 이에만 초점을 맞춰서 제안 과정을 설명하고자 한다.

Approach – Step 2

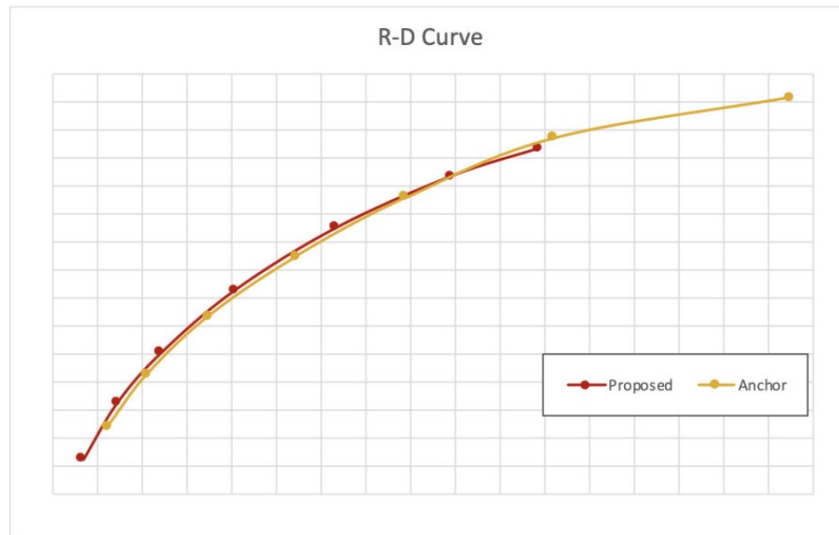
Step 2 (essential)

Scaling을 중간 과정에 포함시킨다. 블록도는 다음과 같다.



Downscaling 과정을 통해 해상도를 낮추고, Super Resolution 또는 Bicubic Interpolation과 같은 기법을 통해 Upscaling을 하여 결과를 얻는다.

이 과정을 통해 R-D Curve를 확인해보면 아래와 같을 것으로 예상된다.



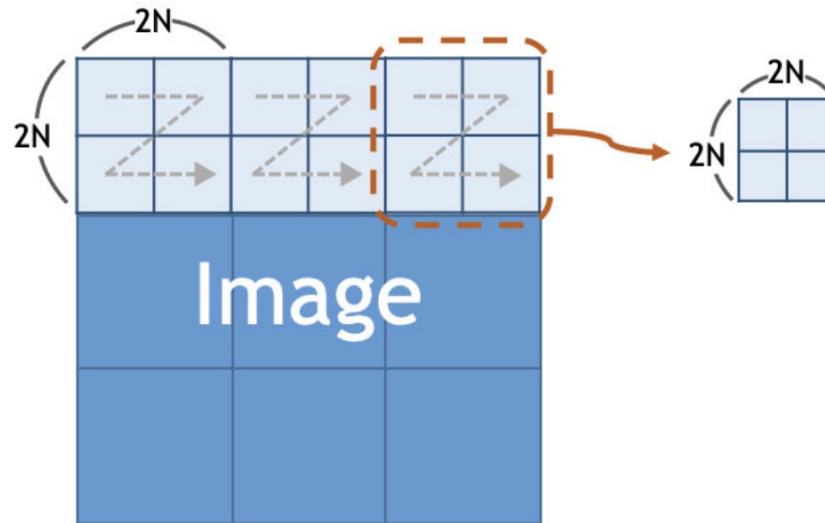
낮은 bpp에선 Anchor보다 PSNR 성능이 좋게 나오지만, 높은 bpp에선 Anchor보다 더 낮은 성능을 보일 것이다. 즉, 저해상도에선 더 좋은 성능을 보이지만 고해상도에선 더 낮은 성능이 예상된다.

Approach – Step 3

Step 3 (additional)

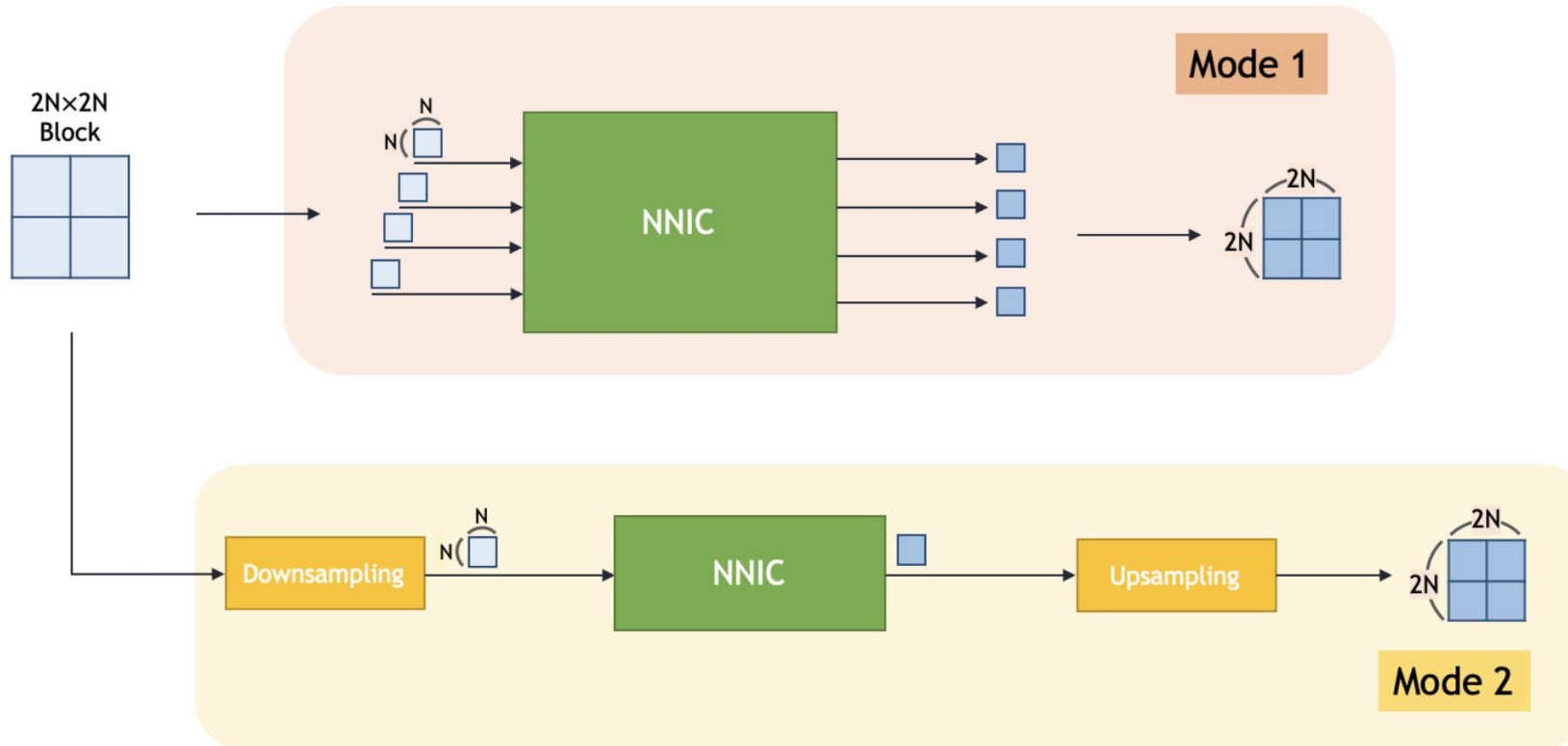
Step 2의 방식처럼 Scaling을 하게 되면 복잡도로 인해 하드웨어 처리에 불리해지고 bitrate에 따른 성능 변화가 생기므로 **블록을 기반으로** 처리하는 방안을 제안한다.

먼저 Image를 여러개의 $2N \times 2N$ 블록으로 나눈다.



Approach – Step 3

Mode 1과 Mode 2, 두가지 모드로 실험을 진행한다.



먼저 **Mode 1**에서는 $2N \times 2N$ 블록을 $N \times N$ 블록 4개로 나누어 $N \times N$ 블록을 입력으로 받는 NNIC에 통과시켜 복원된 블록을 얻는다. 이를 다시 $2N \times 2N$ 블록으로 복원하여, 복원된 $2N \times 2N$ 블록(하늘색)과 오리지널 $2N \times 2N$ 블록(파란색)으로 성능을 측정한다.

Mode 2에서는 $2N \times 2N$ 블록을 Downsampling 하여 $N \times N$ 블록으로 만든다. 이 블록을 $N \times N$ 블록을 입력으로 받는 NNIC에 통과시켜 복원된 블록을 얻고, 이를 $2N \times 2N$ 블록으로 Upsampling 한다. 복원된 $2N \times 2N$ 블록(하늘색)과 오리지널 $2N \times 2N$ 블록(파란색)으로 성능을 측정한다.

각각 Mode에서 입력이 이미지일 때와 블록일 때의 성능을 비교하고, mode끼리의 성능 또한 비교한다.

Approach – Step 4

Step 4 (Challenge)

Step3 단계에선 Down(Up)sampling을 bicubic interpolation 등 hand-craft method를 통해 진행한다. 이번 단계에서는 Sampling(Scaling)을 **NN based Down/Up Scaling**을 통해 진행한다. NN based Down/Up Scaling과 NNIC를 joint training 시켜서 더 좋은 성능을 도출해낼 수 있도록 한다.

Q & A