Super-Resolution based Video Coding Scheme

Hyun min Cho, Kiho Choi
School of Computing, Gacheon University
1342 Seongnamdaero, Sujeong-gu, Seongnam-si, Gyeonggi-do, Korea
cho.hyun@icloud.com aikiho@gachon.ac.kr

Abstract

In this paper, we present a super-resolution-based video coding scheme that compresses video data by combining traditional hybrid video coding and Convolutional neural network-based video coding. During video encoding, downsampling reduces the resolution of an original video in both horizontal and vertical directions to reduce original video data, and Convolutional neural network-based super-resolution is employed after the decoding process to recover the resolution of the reconstructed video during upsampling. For core encoding and decoding processes, the latest video coding standard (i.e., VVC/H.266) is conducted. The experimental results show that the proposed method can provide efficient coding performance while maintaining good visual quality.

1. Introduction

Video coding technology has been constantly evolving for a long time, beginning with the development of media. The importance of video coding technology has begun to be imprinted on the public as a world that prioritizes non-face-to-face communication becomes more prevalent [1].

Video coding technologies have traditionally been developed by standard organizations such as the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). Both organizations produced MPEG-1, MPEG-2/H.262, AVC/H.264, HEVC/H.265, VVC/H.266, MPEG-5 EVC, and other formats, either together or individually [2]. Standard video coding has had a big impact on the broadcasting industry especially and is now used in a lot of services.

The growth of Internet streaming and the over-the-top (OTT) industry has recently been a major topic in the media world. Many individuals are shifting away from traditional broadcasting systems and instead using streaming platforms to watch videos. Although AVC/H.264 and HEVC/H.265 were widely used in the early days of service, AV1 codecs developed by Alliance for Open Media (AOM) have been quickly embraced for OTT service [3].

All conventional standard codecs, as well as AV1, use a hybrid coding method based on signal processing technologies. Video coding approaches based on signal processing have improved and maintained coding performance over time, but they have lately began to show limitations in terms of performance improvement. Researchers have recently sought to overcome the coding performance restrictions of stagnating performance by using Neural Network (NN)-based video coding technology [4].

In this paper, we propose a super-resolution-based video coding scheme that compresses video data by combining a traditional hybrid video coding and a NN-based video coding.

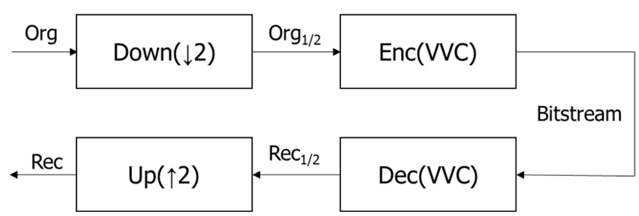

Figure 1: Framework of the proposed method.

2. Super-Resolution based Video Coding

Figure 1 depicts the overall framework of the proposed super-resolution-based video coding scheme. The proposed method is divided into two parts: 1) downsampling and upsampling processes, and 2) encoding and decoding processes. To begin, the proposed scheme employs downsampling, which reduces the resolution of an original frame in both horizontal and vertical directions. Lanczos interpolation filter is used for downsampling, which reduces the resolution by half. For example, if a resolution of original video is 2K, the downsampled resolution is 1K. Following that, VVC/H.266 is used for both encoding and decoding. During the encoding process, a bitstream is generated. Finally, the reconstructed video can be upsampled to its original size.

2.1. VVC based video coding

We used the latest video coding standard known as VVC/H.266 for the encoding and decoding processes. According to the MPEG verification test results [5], VVC demonstrated significantly improved coding performance compared to the previous video coding standard HEVC, resulting in a 50% bitrate reduction. VVC/H.266 is the best regular codec for encoding and decoding processes within the framework of the proposed method. In particular, we used VVenc [6] for encoding and VVdec [7] for decoding. VVenc and VVdec are both opensource VVC implementations that are widely used in academia. VVenc provides

VVenc has five presets for varying encoding speeds/compression quality offsets: faster, fast, medium, slow, and slower. While the slower preset provides the compression efficiency of the VVC reference software (i.e., VTM), the other presets offer highly appealing alternative trade-offs at substantially lower runtimes. Furthermore, the encoder offers single-pass and two-pass rate control modes with multi-threaded encoding, as well as an adaptive quantization mode for optimizing perceptual visual quality while taking into account the human visual system. In the proposed method, we used the slower preset, two-pass coding but did not apply adaptive quantization.

2.2. Super-resolution based video coding

Before the encoding process, downsampling is applied to reduce the resolution ratio in the horizontal and vertical dimensions by a factor of two. For the downsampling, we conducted Lanczos interpolation filter, which is widely used in the resolution change technique for a video. Specifically, mp4 files were converted to YUV files, which were then downsampled using Lanczos interpolation filter in ffmpeg. The downsampled YUV files were passed into VVenC for bitstream generation after the downsampling process.

For the upsampling process, we applied an efficient sub-pixel convolutional neural network (ESPCN)-based SR module to the reconstructed YUV files generated by the VVdeC output. Wenzhe Shi et al. introduced ESPCN in 2016 [8], and the authors reported that the ESPCN has advantages in computational complexity with a small number of model parameters by incorporating the use of an efficient sub-pixel convolution layer to up-sample at the end of the model. Given that video decoding must produce reconstructed frames at the specified frame rate, the use of a low-complexity based SR method should be a mandatory requirement for the NN-based video coding scheme.

To apply ESPCN to the proposed method, the DNN super-resolution module in OpenCV [9] was used. Because the provided ESPCN in OpenCV is trained with BGR format, we converted YUV frames to BGR format and fed them to the ESPCN network. The BGR frame was upsampled in OpenCV using a pretrained ESPCN model, and the upsampled BGR frame was converted back to YUV format. To convert between BGR and YUV formats, we used OpenCV conversion function. The proposed method's upsampling process is depicted in Figure 2, where **X** is the reconstructed frame decoded from VVdeC, and $\mathbf{X}^{\mathbf{SR}}$ is the upsampled reconstructed frame converted back to YUV after the ESPCN upsampling process.

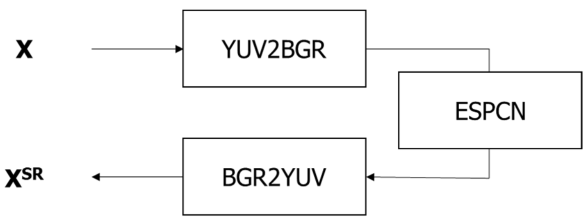

Figure 2: Diagram of SR structures.

3. Experimental result

Table 1 shows the results of the proposed super-resolution-based video coding scheme when tested the method with the validation set provided in the CLIC 2022 video compression track. As shown in the table, the proposed method achieved 28.088dB with 37125820bytes for the 1Mbps category and 25.777dB with 3713339bytes for the 0.1Mbps category, respectively. The decoder size, which included VVdeC and the ESPCN network, was 64960759 bytes and took approximately 5451 seconds to complete.

Figure 3 shows the visual quality of the proposed method in the 1Mbps and 0.1Mbps categories to the original image in the validate set. As shown in the figure, the image encoded with 1Mbps provides similar visual quality to the original image, while the image encoded with 0.1Mbps provides lower visual quality. Nonetheless, regardless of 1Mbps or 0.1Mbps, the proposed super-resolution-based video coding scheme provides good visual quality overall.

Table1. Testing result of the proposed method

Category	PSNR (dB)	Data Size (Byte)	Decoder Size (Byte)	Decoding Time (Second)
0.1 Mbps	25.777	3713339	64960759	5451
1 Mbps	28.088	37125820	64960759	5452

4. Conclusion

In this paper, we presented a super-resolution-based video coding scheme that compresses video data by combining traditional hybrid video coding and NN-based video coding. The experimental results show that the proposed method can provide efficient coding performance while maintaining good visual quality.

(a) Original (b) 1Mbps (c) 0.1Mbps

Figure 3: Visual quality comparisons, where the first column is the original image, the second is the image encoded with 1Mbps, and the third is the image encoded with 0.1Mbps, respectively

5. Acknowledgement

This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (IITP-2021-0-02067) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2021R1F1A1060816).

References

[1] Cisco Visual Networking Index: Forecast and Methodology. 2016–2021.

[2] Kiho Choi, Jianle Chen, Dmytro Rusanovskyy, Kwang-Pyo Choi, Euee S. Jang. An overview of the MPEG-5 essential video coding standard [standards in a nutshell]. *IEEE Signal Processing Magazine*, 37(3):160–7, 2020.

[3] Ian Trow, AV1: Implementation, Performance, and Application. *SMPTE Motion Imaging Journal*, 129(1):51–56, 2020.

[4] Siwei Ma, Xinfeng Zhang, Chuanmin Jia, Zhenghui Zhao, Shiqi Wang, Shanshe Wang. Image and video compression with neural networks: A review. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(6):1683–1698, 2020.

[5] V. Baroncini and M. Wien, VVC Verification Test Report for UHD SDR Video Content, document. JVET-T2020, ITU-T/ISO/IEC Joint Video Experts Team (JVET), Oct. 2020.

[6] VVenC, Online: https://github.com/fraunhoferhhi/vvenc.

[7] VVdeC, Online: https://github.com/fraunhoferhhi/vvdec.

[8] Wenzhe Shi, Jose Caballero, Ferenc Huszar, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, Zehan Wang. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 1874-1883. 2016.

[9] OpenCV, Online: https://github.com/opencv/opencv.