Special Paper
방송공학회논문지 제27권 제7호, 2022년 12월 (JBE Vol. 27, No. 7, December 2022)
https://doi.org/10.5909/JBE.2022.27.7.1021
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

Neural Network based Video Coding in JVET

Kiho Choi[a)‡]

Abstract

After the Versatile Video Coding (VVC)/H.266 standard was completed, the Joint Video Exploration Team (JVET) began to investigate new technologies that could significantly increase coding gain for the next generation video coding standard. One direction is to investigate signal processing based tools, while the other is to investigate Neural Network based technology. Neural Network based Video Coding (NNVC) has not been studied previously, and this is the first trial of such an approach in the standard group. After two years of research, JVET produced the first common software called Neural Compression Software (NCS) with two NN-based in-loop filtering tools at the 27th meeting and began to maintain NN-based technologies for the common experiment. The coding performances of the two filters in NCS-1.0 are shown to be 8.71% and 9.44% on average in a random access scenario, respectively. All the material related to NCS can be found in the repository of the JVET. In this paper, we provide a brief overview and review of the NNVC activity studied in JVET in order to provide trend and insight for the new direction of video coding standard.

Keyword : NNVC, JVET, NN based video coding

I. Introduction

Versatile Video Coding (VVC)/H.266[1] was completed in 2020, offering a bitrate that was roughly half that of the previous video coding standard, High Efficiency Video Coding (HEVC) [2], while maintaining an identical level of visual quality. Thanks to the adoption of new coding tools and the flexible partitioning structure, VVC can achieve two primary goals: half the bitrate with the same quality as HEVC and support for a wide range of applications in a single profile, such as high-resolution video, gaming video, screen content video, 360-degree video, HDR contents, adaptive resolution change, and so on.

Nonetheless, the relatively higher en/decoding complexity of VVC in comparison to HEVC is identified as a problem. The main reason is the evaluation process of all possible Coding Unit (CU) partitions using various coding tools. Signal processing based coding tools demonstrate the limitations of potential coding gains, and the adoption of

a) Gacheon University
‡ Corresponding Author : Kiho Choi
E-mail: aikiho@gachon.ac.kr
Tel: +82-31-750-5799
ORCID: https://orcid.org/0000-0002-2869-0440
※ This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (IITP-2021-0-02067) and the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (NRF-2021R1F1A1060816).
· Manuscript October 17, 2022; Revised December 2, 2022; Accepted December 2, 2022.

numerous coding tools and the introduction of flexible partition structures are thought to be an unavoidable choice, even when much complexity is required to improve coding performance.

Meanwhile, many applications have recently been researched in order to bring advancements in Neural Network (NN) technology. Natural language processing and computer vision, for example, can overcome the performance barrier by leveraging the benefits of machine learning. This trend was widely spread across all technological fields, including video coding [3].

Standard organizations, Moving Picture Experts Group (MPEG) and Joint Video Exploration Team (JVET) founded by MPEG and ITU-T, are also paying close attention to the trend of using NN technology and have begun research into Neural Network based Video Coding (NNVC). At 130th MPEG meeting and 19th JVET meeting, two NNVC-related AHGs were established independently, with the same scope of development: 1) end-to-end (E2E) video coding framework and 2) NN in hybrid video coding framework. The two AHGs were later merged into one under JVET, and the merged adhog group mandated the feasibility of NNVC for the potential coding gain over the conventional video coding standard based on signal processing technology [4]. In this paper, the activity of NNVC studied in JVET is overviewed and reviewed in order to provide trend and insight for the new direction of video coding standard.

The remainder of the paper is as follows: the activity on NNVC and the major contributions are reviewed in Section II and Section III, respectively. Section IV examines and evaluates the performance of NNVC technologies, and Section V concludes the paper.

II. Neural Network Based Video Coding in JVET

NNVC is an essential study item in JVET, although there is no specified target for standardization at this time. When the adhog on NNVC was formed under MPEG, associated needs were explored; however, after NNVC was moved to

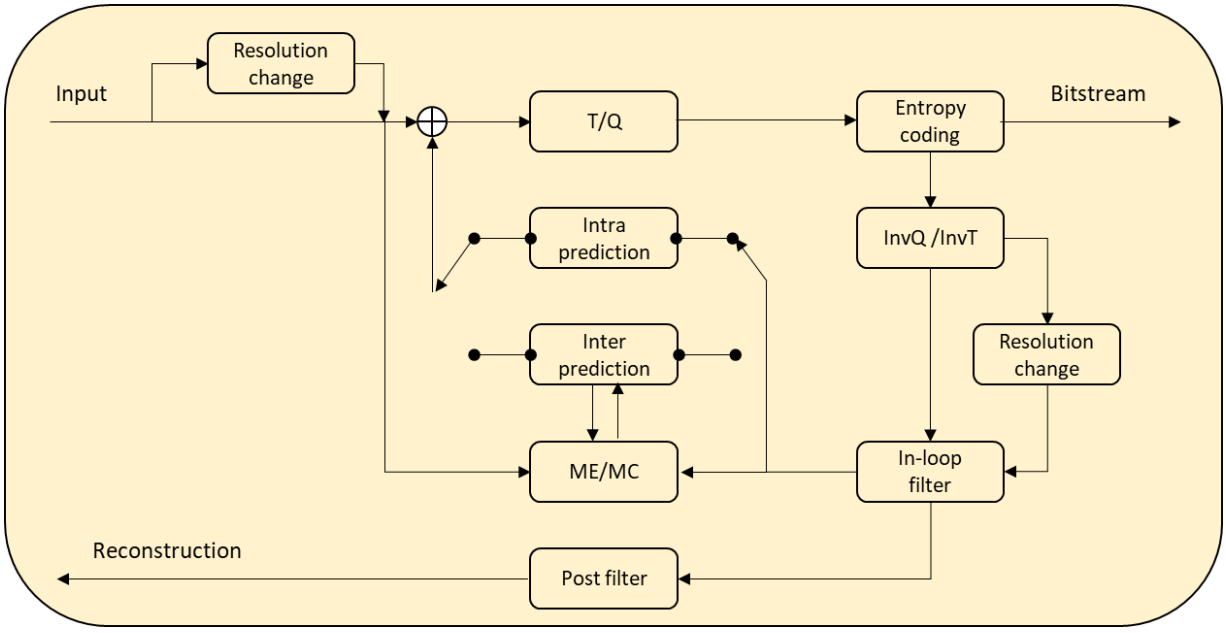


Fig. 1. The example functionality of video coding for NN based coding tool

JVET, NNVC is just researching whether NN-based video coding is promising in terms of coding performance and suitable complexity in encoder and decoder [4].

The JVET focuses primarily on three approaches for NNVC development. The first method employs a neural network in a hybrid video coding framework. The benefit of this approach is that it builds the NNVC technology on the existing hybrid codec architecture. In this approach, the role of NNVC is to build tools by replacing existing tools or adding new tools on top of the hybrid codec framework, which has been the basic framework of video codecs for a long time. Figure 1 depicts the example functionalities used in the traditional video coding framework. NN base coding tool may be in charge of one of functionalities in the framework. For example, a NN based in-loop filter can replace existing in-loop filters in the standard or be added on top of existing in-loop filters.

The second approach is to change video resolution using NN based super-resolution. This approach does not modify a codec framework itself, but rather changes the video resolution for encoding and decoding. Thus, in this approach, existing video coding standards and/or new video coding techniques are used unchanged. Figure 2 shows an example framework. As shown in the figure, the encoder and decoder use conventional video coding standards (e.g., VVC), but NN technology can be used in pre-processing before the encoder and post-processing after the decoder. Super-resolution utilizing NN is traditionally investigated extensively to increase the image resolution in computer vision applications, and rather astonishing results are showed these days. Interestingly, some NN based super-resolution technologies can be utilized directly in conventional video coding. Specifically, if the input size of video resolution is reduced before encoding, the NN based super-resolution method can be applied directly after decoding to recover the reduced resolution. The benefit of using NN based super-resolution in video coding is that we can encode reduced video resolutions in the encoding stage with much smaller bitstream size. As shown in Figure 2, if the pre-processor reduces the input resolution by a factor of two times horizontally and vertically, the encoding data is already reduced by four times at the start of encoding. Using the reduced video data, a conventional video encoder generates bitstreams, and a conventional video decoder generates reconstructed video. Following decoding, the reconstructed video can be up-sampled using the NN based super-resolution method.

The third approach is to develop E2E based NN video coding technology. JVET, similar to NN based technologies investigated in other applications, is also researching E2E video coding technology, which is completely different from the framework of conventional video coding. The first and second approaches would be a compromise between an existing conventional codec framework and NN based technology; however, the third approach would be to use entire chains of NN. This is the primary distinction between this approach and others. Figure 3 depicts the example framework studied in JVET. As shown in the figure, all of the video coding functionality is made up of NN based modules.

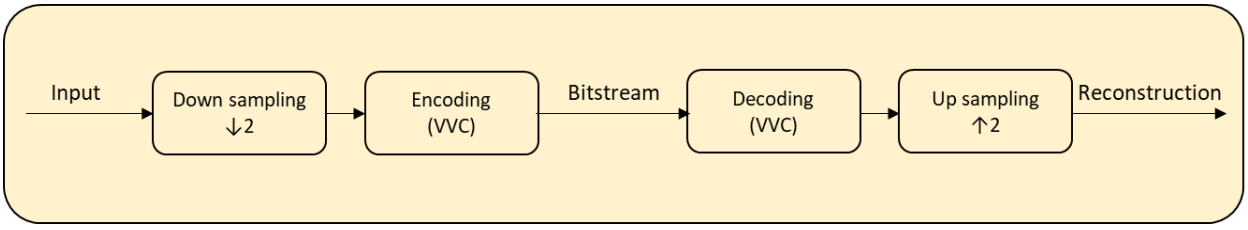


Fig. 2. The example framework for NN based super-resolution

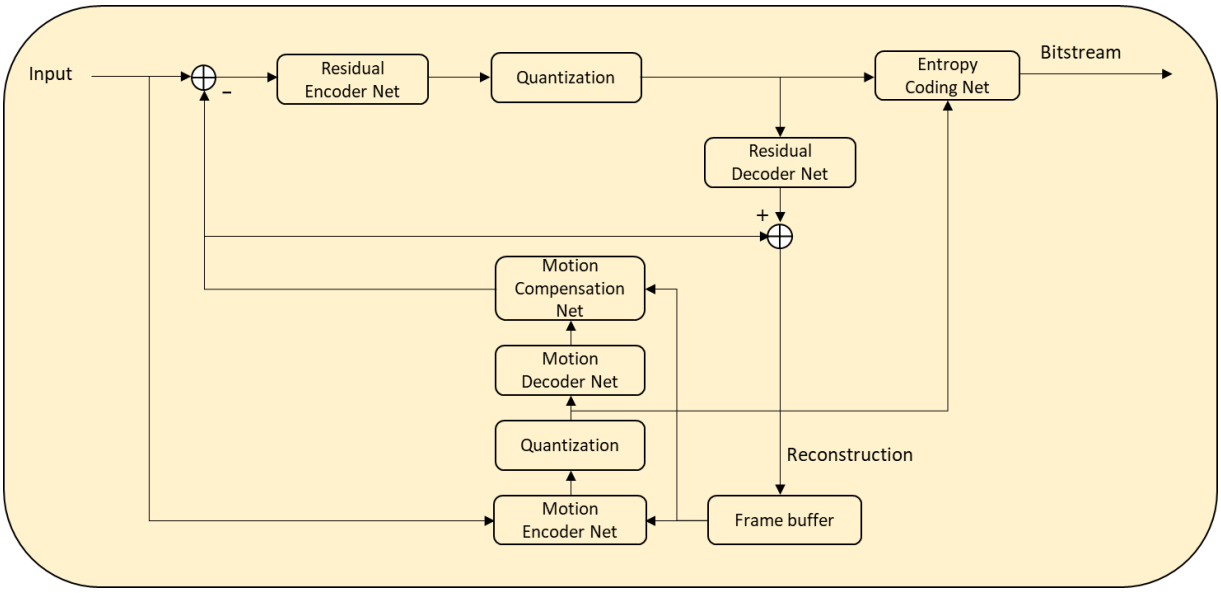
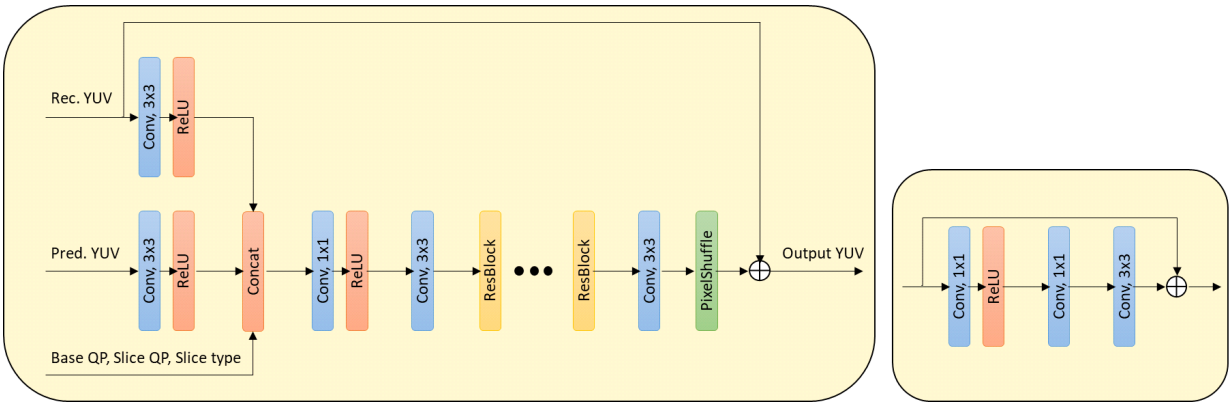


Fig. 3. The example framework for E2E based NN approach

Ⅲ. Review on NNVC contributions in JVET

Table 1 shows the NN related contributions to the 27th JVET meeting, organized by technology area. Since the JVET 27th meeting, numerous NNVC-related contributions have been submitted in three categories: NN-based coding tool, NN-based super-resolution, and End-to-End based NN video coding; it would be difficult to review them all. Furthermore, despite numerous contributions related to NN technologies, there was no meaningful output of the NN technology prior to the JVET 28th meeting. As a result, it is acceptable that this section will focus on reviewing representative contributions summited at the JVET 27th meeting according to the three approaches in order to properly understand the three approaches mentioned in Section II.



(a) (b)

Fig. 4. Network architecture in JVET-AA0088: (a) Overall architecture and (b) ResBlock module

Table 1. NNVC related contributions in JVET 27th meeting

Category	Contribution
NN based in-loop filter	JVET-AA0059, JVET-AA0081, JVET-AA0085, JVET-AA0087, JVET-AA0088, JVET-AA0111, JVET-AA0122, JVET-AA0074, JVET-AA0089, JVET-AA0090, JVET-AA0094, JVET-AA0112, JVET-AA0113, JVET-AA0115. JVET-AA0131. JVET-AA0080
NN based in post filter	JVET-AA0066, JVET-AA0054, JVET-AA0055, JVET-AA0056, JVET-AA0067, JVET-AA0083, JVET-AA0100, JVET-AA0101, JVET-AA0145
NN based Super-resolution	JVET-AA0071, JVET-AA0065, JVET-AA0076, JVET-AA0084
End to End NN	JVET-AA0063
Inter prediction	JVET-AA0082
Software	JVET-AA0086

1. Contribution related to NN based coding tool

The majority of contributions, as shown in Table 1, are related to NN based filters. The first contribution to review is one of the NN based coding tool contributions, JVET-AA0088 [5], which proposes a NN based in-loop filter that can be used to replace the deblocking and SAO filters. The proposed a convolutional neural network (CNN) filter is trained separately for I and B slices in the contribution to improve the quality of the reconstructed image. Specifically, the proposed CNN filter uses the reconstructed image after LMCS as the input of network, and the output of CNN filter is processed by ALF and CCALF. In order to feed YUV to a filter model, the U and V channels are up-sampled and down-sampled in pre-processing and post-processing, respectively.

The architecture and residual block (ResBlock) module of the proposed method in [5] is shown in Figure 4. The basic overall architecture consists of connecting ResBlocks to refine the input frame of reconstructed YUV. The ResBlock is made up of three convolutional layers. The first layer in the ResBlock is a 1x1 convolutional layer followed by a ReLU activation function, the second layer is also a 1x1 convolutional layer, and the third layer is a 3x3 convolutional layer. The number of feature maps for the internal convolutional layers is set to 64. The network is fed reconstructed YUV, prediction YUV, base QP, slice QP, and slice type as input values for overall architecture. In this contribution, the number of ResBlocks linked is set to 8.

The main highlights of the proposed method in the contribution can be summarized by three features. Firstly, the proposed method can replace existing deblocking and SAO filters, thereby covering two functions that were previously used in conventional video codecs. The second feature of the proposed method is that it can be used adaptively at the CTU and slice levels. This adaptivity can provide more enhanced reconstructed video quality depending on the characteristics of the contents. Thirdly, the proposed method includes a scaling operation, which is used to refine the output of a NN filter. The slice header specifies the scaling factors for each component.

The second NN based coding tool contribution is JVET-AA0111 [6], which proposes a deep in-loop filter with adaptive parameter selection. To improve the coding efficiency of VVC, a deep in-loop filter with an adaptive parameter selection mechanism and an external attention based architecture is proposed in the contribution. The proposed CNN model investigates reconstruction, partitioning, boundary strength, and QP as input, and residual scaling (i.e., the differences between the input samples and the NN filtered samples) is further exploited before being added to input samples.

Furthermore, the proposed method investigates the combination of the proposed method and the deblocking filter, so the method uses the input samples used in the residual scaling from the output of deblocking filtering.

$$F\_out = F\_in \ f(Rec, Pred, BS, QP) + F\_in \qquad (1)$$

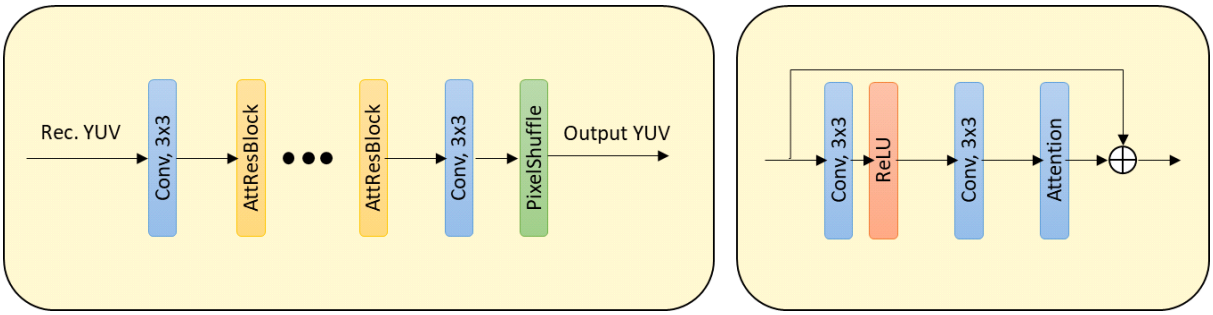


Fig. 5. Network architecture in JVET-AA0111: (a) Overall architecture and (b) Attention Residual Block module

where F_in and F_out represent the input and the output of the attention module in Figure 5, and Rec, Pred, BS, and QP represent the reconstruction, the prediction, the boundary strength, and the input quantization parameter, respectively. As shown in Figure 5. (b), f consists of two convolutional networks, and an activation function is applied in between the two convolutional networks. The goal of f is to generate a spatial attention map from external data, which can be used to recalibrate the feature maps F_in. In the overall architecture, the numbers of feature maps and attention residual blocks are cascaded with specific the number of feature maps and the number of samples in one dimension, such as 96 and 8 in the 8-residual block version and 96 and 16 in the 16-residual block version.

One of the most appealing aspects of the proposed method in [6] is the technique known as parameter selection. The proposed method can determine whether the CNN filter is applied at the slice or CU block level. When the CNN filter is determined to be applied at the slice or block level, the CNN filter need to choose one of the parameters from the QP based candidate list. The list includes three sequence level conditional parameters: QP, QP-5, and QP-10. The selection procedure is based on the normal rate-distortion optimization process, which employs rate-distortion cost in the encoder. The on/off information of the CNN filter and index of the selected conditional parameter can be signaled to the decoder.

The two contributions, JVET-AA0088 and JVET-AA0111, examined for NN based coding tools demonstrate good coding performance; both contributions were adopted in the initial version of NNVC common SW, which will be maintained for the NN technology.

2. Contribution related to NN based super-resolution

JVET-AA0071 [7] can be exampled in NN based super-resolution contribution, which utilizes a convolutional network used as up-sampling filter for Y component to improve the compression efficiency. This contribution consists of two parts to change of video resolution. The first part is a method on GOP level encoding resolution decision and the second part is CNN based super-resolution method.

Regarding the first part, the original contribution was proposed in JVET-Z0065 [8]. The key technology in the contribution is a change in input resolution at the GOP level. For example, the resampling factors between half size and original size are selected adaptively at the GOP level, and the selected scale factor is applied to all frames in the GOP. To determine whether down sampling as half size is required or not at the GOP level, the PSNR is calculated between the original picture and the down-up scaled picture, where the first picture is downscaled to quarter resolution and then resampled to the original resolution. After

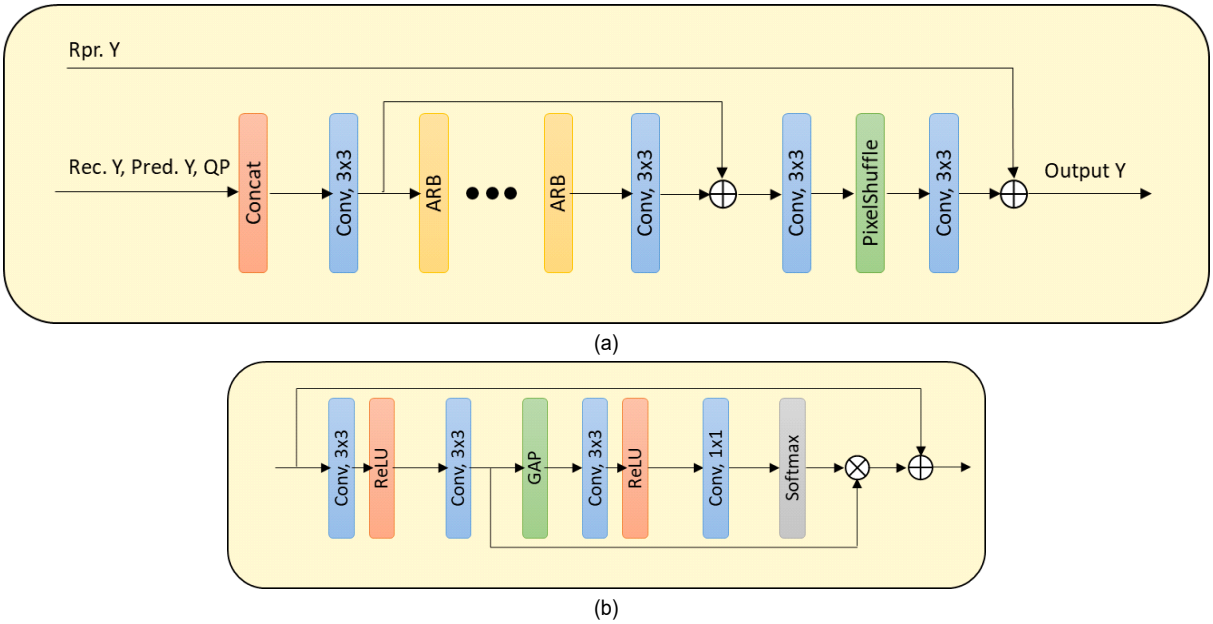


Fig. 6. Network architecture in JVET-AA0071: (a) Overall architecture and (b) Advanced Residual Block module

comparing the PSNR number to a threshold stored in the pre-defined table, the scale factor is chosen based on a threshold. The thresholds are calculated using the initial QP. Such the role is also applied to JVET-AA0071.

The second part, CNN based super-resolution method proposed in JVET-Z0088 [9], is described in Figure 6. As aforementioned, the CNN network is designed for Y component. Specifically, the overall architecture consists of sixteen advanced residual blocks (ARB), four convolutional layers, a concatenate layer, and a shuffle layer. For the efficient training, shortcut connection in between convolutional layers and a global connection from input to output are included. The parameters of each convolution layer, similar of other CNN models, are [cin, k, cout], where cin, k, and cout represent the number of input channels, the size of the convolution kernel, and the number of output channels, respectively. The input of CNN model is as follows: QP, YRpr, YLR-Rec, YLR-Pred, and YLR-Pred where YRpr represents reconstruction up-sampled by RPR, YLR-Rec represents reconstruction, YLR-Pred represents prediction, and QP represents quantization parameter. The output of model is the super-resolution reconstruction YSR. The architecture of the designed ARB is shown in Figure 6 (b), where ReLU, GAP, and Softmax are the activation function, global average pooling layer, and normalization function, respectively.

The basic framework for NN based super-resolution can be examined in this contribution by changing the resolution before encoding and after decoding. Essentially, it is believed that the factor of ratio for down-sampling and up-sampling, as well as synchronization between down-sampling and up-sampling processes, is critical for improving coding efficiency. Because the coding performance of NN based super-resolution can vary depending on the video contents, the critical factors should be investigated further.

3. Contribution related to End-to-End based NN video coding

JVET-AA0063 [10] is considered as an E2E based NN video coding technology contribution. The contribution is

an extension version of the E2E based NN video coding technology for omnidirectional videos proposed in JVET-X0043 [11] and JVET-Y0051 [12]; thus, the basic architecture is the same as that of the omnidirectional videos proposed in [11] and [12].

The overall encoding framework of the proposed method in JVET-AA0071 is depicted in Figure 7 and 8. As shown in the figures, all of the modules use NN, and each module is linked to the others to form a full chain network. Projection, separate channels, bidirectional motion estimation, motion encoder/decoder, bidirectional motion compensation, residual encoder/decoder, quality enhancement, and entropy coding are the major modules in the architecture. Interestingly, even if all of the modules are implemented using neural networks, such modules in the proposed method follow a simplified functionality of the architecture of hybrid video coding. When comparing the proposed method with the NN based tool architecture in Figure 1, the essential functionality is similar to each other.

One thing to note about this contribution is that the proposed E2E-based NN video coding method is just for inter frames. For intra frame (i.e., I frame), the proposed method still employs traditional video coding standards, such as VVC intra coding or BPG compression. Given the fact that

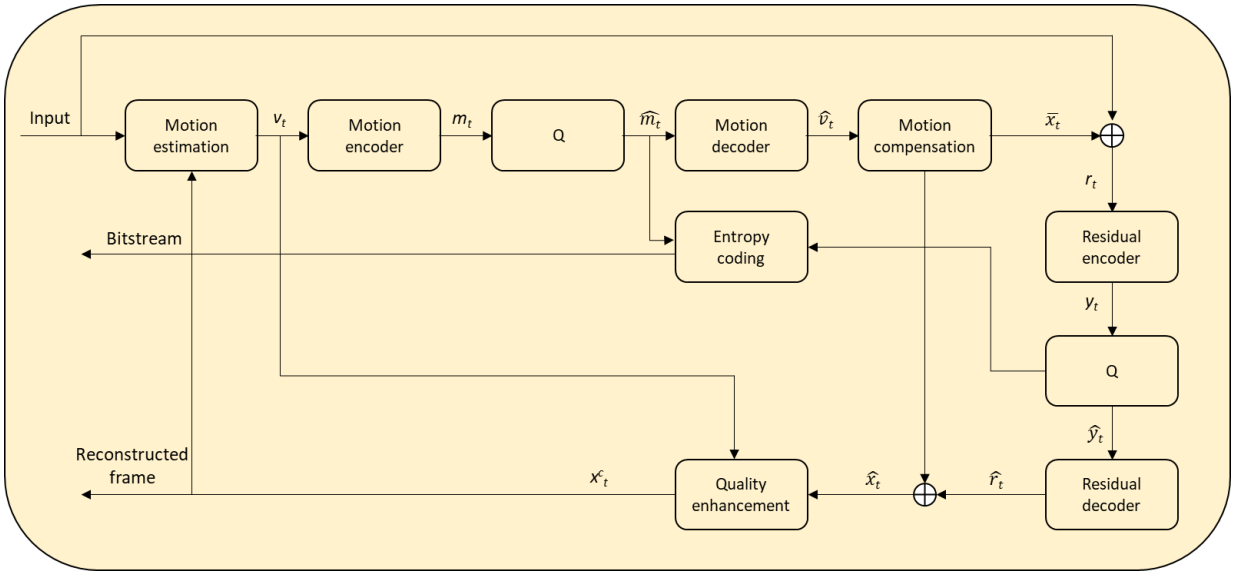


Fig. 7. Overall framework of the encoder in JVET-AA0063

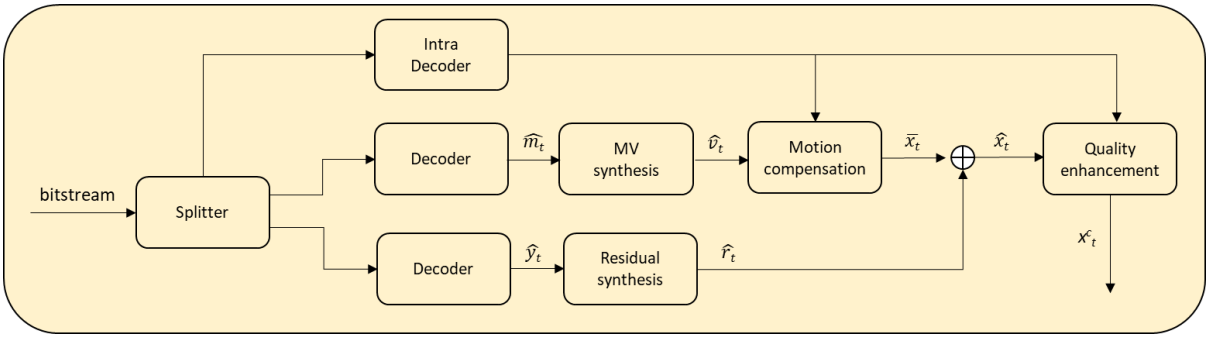


Fig. 8. Overall framework of the decoder in JVET-AA0063

E2E image coding is extensively studied recently, it is expected that it would not be difficult to exploit intra coding based NN on top of the E2E inter coding framework.

In contrast to conventional signal processing based video coding, the E2E concept appears to be destructive. According to recent JPEG reports, compression capability is even outperforming the most recent intra coding (e.g., VVC intra coding) in image coding. However, it seems that E2E-based NN video coding takes more time to get a reasonable coding performance. Currently, most E2E-based NN video coding technologies in JVET showed a worse coding performance than that of signal processing based framework, and this contribution also performed worse than VVC for inter coding. The detail number will be provided in Section IV.

Ⅳ. Experimental results

For NNVC testing, JVET setup a common test conditions and evaluation procedures [13]. In the document, there is a specific test scenario, sequences, and metrics. Specially, the document defines training procedures, conversion practices and software reference configurations in the context of NNVC experiments.

1. Performance evaluation

In order to check the coding performance of NN technology, the contributions analyzed in Section II are evaluated according to the CTC document. Table 2 shows the summary of coding performance compared to the anchor. The Bjntegaard delta bitrates (BDBR) [14] were used to evaluate the coding performance of NN technology to VTM11.0 [15] that is anchor of the test.

As shown in the table, JVET-AA0088 shows the coding improvements of RA scenario in Y, Cb, and Cr coding of 8.7%, 19.4%, and 19.5% on average, and the coding improvements of AI scenario in Y, Cb, and Cr coding of 6.5%, 15.1%, and 15.8% on average, respectively. JVET-AA0111 shows the coding improvements of RA scenario in Y, Cb, and Cr coding of 9.8%, 22.3%, and 22.8% on average, and the coding improvements of AI scenario in Y, Cb, and Cr coding of 7.3, 20.4%, and 21% on average, respectively. JVET-AA0071 shows the coding improvements of RA scenario in Y, Cb, and Cr coding of 1.5%, -1.0%, and -1.1% on average, and the coding improvements of AI scenario in Y, Cb, and Cr coding of 1.2%, -1.4%, and -1.3% on average, respectively. JVET-AA0071 shows the coding loss of RA scenario in Y, Cb, and Cr coding of 4.3%, 4.0%, and 3.7% on average, respectively.

NN based tool improvements for in-loop filter show relatively good coding performance in the current state of NNVC. In the RA scenario, both contributions, JVET-AA0088 and JVET-AA0111, shows approximately 10%. This improvement is quite impressive given that signal processing based tools in VVC only showed a 1-2% improvement. NN based super-resolution technology and E2E based NN technology appear to require further

Table 2. Coding performance of NN based contributions

Contribution	Category	Random Access (CTC)			All Intra (CTC)		
		Y	Cb	Cr	Y	Cb	Cr
JVET-AA0088	NN based in-loop filter	-8.7%	-19.4%	-19.5%	-6.5%	-15.1%	-15.8%
JVET-AA0111	NN based in-loop filter	-9.8%	-22.3%	-22.8%	-7.3%	-20.4%	-21.0%
JVET-AA0071	NN based super-resolution	-1.5%	1.0%	1.1%	-1.2%	1.4%	1.3%
JVET-AA0059	End to End NN	4.3%	4.0%	3.7%	N/A	N/A	N/A

improvement. Although there is a small improvement in Y for NN based super-resolution technology, coding loss still appears in Cb and Cb components. Even E2E based NN technology shows coding losses in all components of about 4% in the RA scenario when compared to VTM11.0. The fundamental reason for the low coding performance is that the two approaches, NN-based super-resolution and E2E-based NN technology, have not been completely explored; however, it is believed that good coding gain will be showed if additional study is undertaken.

2. Complexity evaluation and analysis

To assess the complexity of NN based technology, two measurements are used: operational complexity and time complexity. JVET calculates the total number of parameters and the required kMAC number per pixel to determine operational complexity. These measurements can be used to estimate the amount of memory and processing time needed in a practical implementation. In terms of time complexity, the runtimes of encoding and decoding are compared to anchor. The relative time of anchor can be calculated using the T measurement T = Ttest / Tanchor * 100%, where Ttest and Tanchor represent the runtimes of the tested method and the anchor method, respectively. A value of 100% indicates that the run-time is the same. For the sake of simplicity, the running time was measured on a CPU platform that did not support GPU configuration.

Table 3 shows the summary of operational complexity and the time complexity compared to the anchor. As shown in the table, JVET-AA0088 requires 1.9 million parameters in total and 485 kMAC/pixel in worse case. Regarding time complexity, the JVET-AA0088 shows the increase of encoding time in RA and AI of 228% and 235% on average, and the increase of decoding time in RA and AI of 211432% and 114004% on average, respectively. JVET-AA0111 requires 6.2 million parameters in total and 649 kMAC/pixel in worse case. Regarding time complexity, the JVET-AA0111 shows the increase of encoding time in RA and AI of 186% and 146% on average, and the increase of decoding time in RA and AI of 43283% and 24974% on average, respectively. JVET-AA0071 requires 2.9 million parameters in total and 361 kMAC/pixel in worse case. Regarding time complexity, the JVET-AA0071 shows the increase of encoding time in RA and AI of 67% and 69% on average, and the increase of decoding time in RA and AI of 291% and 180% on average, respectively. JVET-AA0059 requires 81 million parameters in total. Regarding time complexity, the JVET-AA0059 shows the increase of encoding time in RA of 3% and the increase of decoding time in RA of 173% on average, respectively.

Although NN based super-resolution technology and E2E based NN technology shows faster encoding times than anchor, the overall operational and time complexities

Table 3. Complexity of NN based contributions

Contribution	Category	Operational complexity		Time complexity			
				Random Access (CTC)		All Intra (CTC)	
		Total Number of Parameters (Millions)	Worst Case Complexity (kMAC/pixel)	EncT (%)	DecT CPU (%)	EncT (%)	DecT CPU (%)
JVET-AA0088	NN based in-loop filter	1.9	485	228%	211432%	235%	114004%
JVET-AA0111	NN based in-loop filter	6.2	649	186%	43283%	146%	24974%
JVET-AA0071	NN based super-resolution	2.9	361	67%	291%	69%	180%
JVET-AA0059	End to End NN	81.0	N/A	3%	173%	N/A	N/A

appear to be very high. Because the required parameters and pixel operations are impractical for decoder implementation, and the increased decoding time may be impractical for real-time processing, the complexity of NNVC appears to require extensive research. With regard to the trade-off calculation between operational complexity and coding performance, NN-based in-loop filters show a good coding performance with increased complexity, whereas NN based super-resolution technology and E2E based NN technology show poor trade-off between coding performance and complexity in the current state of NNVC activity.

Ⅴ. NNVC common software

In the JVET 27th meeting, JVET decided to develop a common NNVC software called Neural Compression Software (NCS). Throughout the two years of research, there were constant voices to develop common test software. NCS was established for a variety of reasons. The first reason is that despite two years of exploration experiments, there was no meaningful output from the NN technologies because the group did not use a common software to maintain NN technologies. Second, there was no identical test bed that could operate under identical testing conditions. The same testbed should be supported to evaluate the proposed method, as this is a reliable way to check the actual coding performance. Finally, there was no detailed guideline providing examples of training and testing for NN technologies such as data dumping, data generation, training, examples of implementation for inference and signaling, and so on.

As a result, in the 27th JVET meeting, JVET selected two NN-based technologies that are well-studied loop filter packages that are considered mature enough to be included in common software code bases, and ported the two NN-based in-loop filtering tools on top of VTM11.0, and released NCS-1.0 [16]. The two tools chosen are the JVET-AA0088 and JVET-AA0111 in-loop filtering tools introduced in 3.1. The following features are included in NCS-1.0:

- Common API for data dumping and loading
- Common API for inference
- Common part for SPS
- NN filter sets: Set0 (filter proposed in AA0088) and Set1 (filter proposed in AA0111)
- Training scripts for each set: Data dumping, Dataset generation, Training, Conversion to SADL format

Detailed information for training and testing the two NN-based in-loop filtering tools on the VTM11.0 can be found in documents in [16].

Table 4 shows the coding performance of NN based filtering tools adopted in NCS-1.0. Anchor of the test is VTM-11_nnvc-2.0 [17] and the testing version of the sets is set to test the version of int16 precision implementation. As shown in the table, Set0 achieves 8.7%, 7.9%, and 6.5% bitrate savings in the RA, LDB, and AI configurations on average, and Set1 achieves 9.4%, 8.5%, and 7.3% bitrate savings in the RA, LDB, and AI configurations on average. Given that the majority of signal processing-based coding tools in VVC show less than 1%, the coding performance

Table 4. Coding performance of NN based filter sets in NCS-1.0

#Set	Random Access			Low Delay B			All Intra		
	Y	Cb	Cr	Y	Cb	Cr	Y	Cb	Cr
Set0	-8.7%	-18.2%	-18.9%	-7.9%	-18.7%	-20.2%	-6.5%	-15.5%	-16.6%
Set1	-9.4%	-20.7%	-20.4%	-8.5%	-15.6%	-14.4%	-7.3%	-20.1%	-20.6%

of Set0 and Set1 showed a significantly improved number in an individual tool.

Ⅶ. Conclusion

In this paper, we reviewed the NNVC activity studied in JVET, which has been started to investigate NN based video coding technologies for the next generation video coding standard. The NNVC is studying three approaches using NN advances majorly as 1) NN based coding tool, 2) NN based super-resolution technology, and 3) E2E based NN video coding technology. Especially, NN based filters are extensively studied and showed relatively good coding performance in the experimental results. In conclusion, the research on NNVC is just started, and it is time to see whether such the direction is promising for the success of future video coding standard.

References

[1] Versatile Video Coding, Recommendation ITU-T H.266 and ISO/IEC 23090-3 (VVC), ITU-T and ISO/IEC JTC 1, Jul. 2020.

[2] High Efficiency Video Coding, Recommendation ITU-T H.265 and ISO/IEC 23008-2 (HEVC), ITU-T and ISO/IEC JTC 1, Apr. 2013.

[3] S. Ma, X. Zhang, C. Jia, Z. Zhao, S. Wang, S. Wang, “Image and video compression with neural networks: A review.” IEEE Transactions on Circuits and Systems for Video Technology, 2019. doi: https://doi.org/10.1109/TCSVT.2019.2910119

[4] A. Alshina, S. Liu, J. Pfaff, M. Wien, P. Wu, Y. Ye, “JVET AHG report: Neural-network-based video coding (AHG11)”, JVET-T0011, Oct. 2020.

[5] L. Wang, S. Lin, X. Xu, S. Liu (Tencent), F. Galpin (InterDigital), “EE1-1.5: Neural network based in-loop filter with a single model”, JVET-AA0088, Jul. 2022.

[6] Y. Li, K. Zhang, J. Li, L. Zhang (Bytedance), H. Wang, M. Coban, A.M. Kotra, M. Karczewicz (Qualcomm), F. Galpin (InterDigital), K. Andersson, J. Ström, D. Liu, R. Sjöberg (Ericsson), “EE1-1.6: Deep In-Loop Filter With Fixed Point Implementation”, JVET-AA0111, Jul. 2022.

[7] S. Peng, C. Fang, D. Jiang, J. Lin, X. Zhang (Dahua), J. Nam, S. Yoo, J. Lim, S. Kim (LGE), “EE1-2.1: A CNN-based Super Resolution Method Combined with GOP Level Adaptive Resolution”, JVET-AA0071, Jul. 2022. https://jvet-experts.org/doc_end_user/documents/27_Teleconference/wg11/JVET-AA0071-v2.zip

[8] J. Nam, S. Yoo, J. Lim, S. Kim (LGE), “EE1-2.1: RPR encoder with multiple scale factors”, JVET-Z0065, Apr. 2022.

[9] S. Peng, D. Jiang, J. Lin, C. Fang, X. Zhang (Dahua), “AHG11: A CNN-based Super Resolution Method Combined with Existing RPR Functionality”, JVET-Z0088, Apr. 2022.

[10] Y. He, B. Wang, E. Alshina, J. Sauer, “AHG11: A hybrid codec using E2E image coding combined with VVC video coding”, JVET-AA0063, Jul. 2022.

[11] Qipu Qin, Cheolkon Jung, Zou Dan, Ming Li, “[AHG11 & AHG6] DOVC: Deep Omnidirectional Video Compression”, JVET-X0043, Oct. 2021.

[12] Qipu Qin, Cheolkon Jung (Xidian University), Dan Zou, Ming Li (OPPO), “AHG11: Deep omnidirectional video compression in YUV domain”, JVET-Y0051, Jan. 2022.

[13] E. Alshina, R.-L. Liao, S. Liu, A. Segall, “Common Test Conditions and evaluation procedures for neural network-based video coding technology”, JVET-Z2016, Apr. 2022.

[14] G. Bjøntegaard, “Improvement of BD-PSNR Model”, ITU-T SG16/Q6 VCEG-AI11, Jul. 2008.

[15] VVC Reference Software. https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware _VTM/-/tags/

[16] Neural Compression Software (NCS). https://vcgit.hhi.fraunhofer.de/jvet-ahg-nnvc/VVCSoftware_VTM/-/tree/VTM-11.0_nnvc

[17] JVET Common Test Conditions for Neural Network-Based Video Coding Technology. https://vcgit.hhi.fraunhofer.de/jvet-ahg-nnvc/ nnvc-ctc/-/tree/master

Introduction Authors

Kiho Choi

- 2012. : Electronics and computer engineering, Hanyang University, B.S.(2008), Ph.D.(2012)
- 2012. 09. ~ 2014. 02. : Lecturer, Post Doc., Hanyang University
- 2014. 03. ~ 2021. 02. : Senior Engineer, Visual Technology Lab. of Samsung Research
- 2021. 03. ~ Current : Professor of School of Computing, Gachon University
- ORCID : https://orcid.org/0000-0002-2869-0440
- Research interests : Image and video coding, multimedia data compression, multimedia streaming, AI-based multimedia technology, and multimedia standardization