Prediction-Aware Quality Enhancement of VVC Using CNN

Fatemeh Nasiri, Wassim Hamidouche, Luce Morin, Nicolas Dhollande, Gildas Cocherel

Prediction-Aware Quality Enhancement of VVC Using CNN

Fatemeh Nasiri [⋆†+], Wassim Hamidouche[⋆†], Luce Morin[†⋆], Nicolas Dhollande[+], Gildas Cocherel[+]
[⋆] IRT b<>com, 35510 Cesson-Sévigné, France,
[†] Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, 35000 Rennes, France
[+] AVIWEST, 35760, Saint-Grégoire, France

***Abstract*—The upcoming video coding standard, Versatile Video Coding (VVC), has shown great improvement compared to its predecessor, High Efficiency Video Coding (HEVC), in terms of bitrate saving. Despite its substantial performance, compressed videos might still suffer from quality degradation at low bitrates due to coding artifacts such as blockiness, blurriness and ringing. In this work, we exploit Convolutional Neural Networks (CNN) to enhance quality of VVC coded frames after decoding in order to reduce low bitrate artifacts. The main contribution of this work is the use of coding information from the compressed bitstream. More precisely, the prediction information of intra frames is used for training the network in addition to the reconstruction information. The proposed method is applied on both luminance and chrominance components of intra coded frames of VVC. Experiments on VVC Test Model (VTM) show that, both in low and high bitrates, the use of coding information can improve the BD-rate performance by about 1% and 6% for luma and chroma components, respectively.**

***Keywords*—CNN, Intra VVC, quality enhancement**

I. Introduction

Video streaming applications have gained more popularity in the past few years. Therefore, the task of delivering a high quality video has become essential. From the compression point of view, the upcoming video coding standards, in particular VVC, can achieve up to 50% bitrate saving compared to its predecessor HEVC [1]. Alongside the video coding progress, receiver devices have also become more powerful in processing received videos and enhancing their quality. As a result, video post-processing is nowadays an interesting option for display manufacturers in order to further improve the viewing experience of their users.

The promising performance of machine learning methods has recently encouraged researchers to exploit them in the video compression domain. Particularly, deep Convolutional Neural Networks (CNN) have attracted more attention owing to their significant performance [2], [3]. Despite the interesting performance of CNN-based methods, they usually impose a high computational complexity which makes them unsuitable for real-time encoding applications. However, the post-processing approaches which improve the reconstructed video after the decoding step can be more flexible, since they are not involved in the encoding and decoding process. In other words, Such post-processing approaches can serve as an optional step to be used based on the hardware capacity of the decoder device.

CNN-based quality enhancement (QE) for VVC has been sparsely studied in the literature. The existing works target both intra and inter frames of coded videos. In [4]–[11], CNN-based methods have been proposed to the VVC standardization either as in-loop filter or post-processing step. Considering the fact that the distortion in compressed video is influenced by the encoding process and its decision making engine, an attention based network is proposed in [12], where partitioning information of VVC is exploited to further increase the performance of the QE filter. Finally, in [13], the impact of network architecture complexity on the performance of the QE filter has been studied.

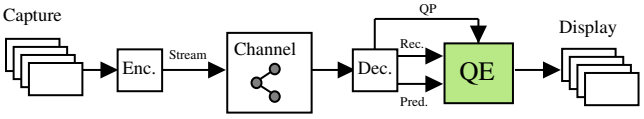

Fig. 1: Compressed video quality enhancement framework

In this paper, a CNN-based QE method is proposed, which follows the objective of the previously presented works with the use of coding information [4], [12], [14]. The main contribution of this work is that we use the spatial predictor of each frame as the input to the CNN. This is motivated by the fact that coding information, such as intra prediction signal, usually represent a useful information about the type of the distortion [15]. Fig 1 presents the overall workflow of the proposed method. The input of the QE neural network is the decoded frame, the intra prediction information and the Quantization Parameter (QP). The CNN architecture of this paper is inspired by the network proposed in [16], which has shown great performance for the super resolution problem. Moreover, the three color components of each frame are processed separately.

The rest of this paper is organized as follows. In Section II, the proposed QE method using intra prediction as coding information is presented. Experimental results as well as discussions and comparisons with state of the art solutions are provided in Section III and finally, Section IV concludes the paper.

II. Predictor-aware quality enhancement

In this section, first we will explain the intuition and motivation for using intra prediction in the proposed CNN-based QE method. Then, network architecture and training configuration will be presented.

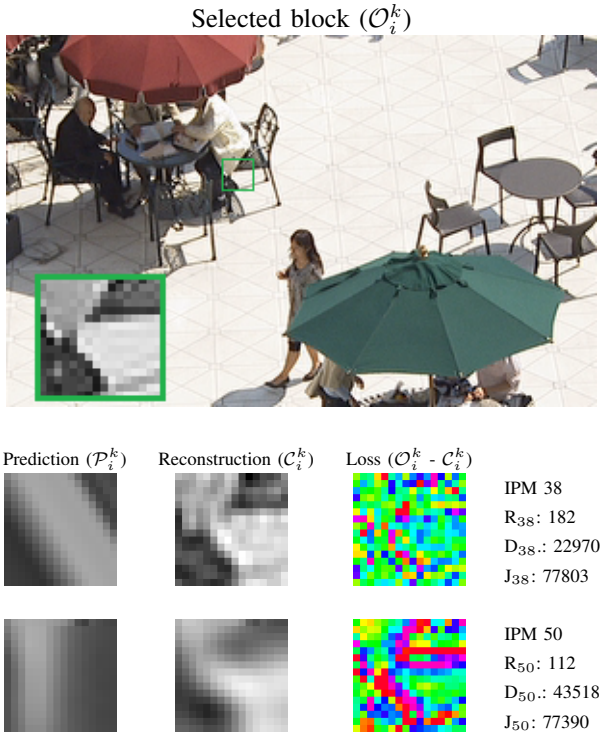

Fig. 2: A 16 × 16 block, *k*, and its two best IPMs (*i* = 38, 50), with similar costs but different rate-distortion trade-offs resulting in distinct compression loss patterns (QP 40: λ=301)

A. *Intra coding and compression artifacts*

In intra coding, each block is predicted based on its neighboring pixels, given some predefined models. In VVC, these models include a set of 67 Intra Prediction Modes (IPM), representing 65 angular IPMs, plus DC and planar. Like other decisions in video coding, the selection of an IPM for a block consists in optimizing a function of the rate and the distortion, called the rate-distortion (R-D) cost. Particularly for intra coding, the the R-D cost of an IPM *i*, denoted as J_i, is computed as

$$J_i = D_i + \lambda \times R_i \qquad i = 1, ..., 67, \tag{1}$$

where D_i and R_i are the distortion and the rate, obtained when using *i* as the IPM of the block, respectively. Moreover, λ is the Lagrangian multiplier, computed based on the QP which determines the relative importance of the rate and the distortion during the decision making process. For instance, in low bitrates (high QP), the value of λ is higher, which indicates that minimization of the rate is relatively more important than minimization of the distortion.

Strict bitrate constraints might cause a situation where the best IPM minimizing the R-D cost of a block, is not necessarily the IPM that models the block texture most accurately. Fig. 2 shows an example of such a situation in the first frame of the BQSquare sequence. In this figure, a 16 × 16 block, *k*, is selected and the Prediction ($\mathcal{P}_i^k$) and Reconstruction ($\mathcal{C}_i^k$) blocks corresponding to its two best IPMs in terms of R-D cost are shown. As can be seen, despite their similar R-D costs, these two IPMs result in very different reconstruction signals, with different types of compression loss patterns. This behavior is due to two different R-D trade offs of the selected modes.

On one hand, IPM 38 is able to model the block content more accurately (i.e. smaller distortion D_{38}) with the cost of a higher IPM and residual coding rate (i.e. R_{38}). On the other hand, IPM 50 provides a less accurate texture modeling (i.e. high distortion D_{50}) with a smaller rate residual and IPM coding rate (i.e. R_{50}). Consequently, these two IPMs result in very different types of artifacts for the given block, as can be seen by comparing the corresponding reconstruction blocks (i.e. $\mathcal{C}_{38}^k$ and $\mathcal{C}_{50}^k$).

The above example proves that the task of QE for a block, frame or an entire sequence could be significantly impacted by different choices of coding modes (e.g. IPM) determined by the encoder. This assumption is the main motivation in our work to use the intra prediction information for training of the quality enhancement networks.

B. *Proposed CNN-based quality enhancement method*

The proposed QE algorithm is applied on intra frames after decoding. In order to accurately capture the compression loss, as explained in previous section, the prediction information is also extracted from the decoder and is used as the input to the QE network. For each reconstruction frame, this prediction information is composed of predictors associated to its blocks. The predictor of each block is the projection of its reference pixels corresponding to the angle of the used IPM. The reconstruction and prediction frames are concatenated and fed to the network as one input image.

Inspired by the architecture of the Enhanced Deep Super Resolution (EDSR) [16], we have exploited residual training in our QE network. The architecture of the QE network is shown in Fig 3. The first convolutional layer receives the reconstruction and prediction frames as input. In the next step, after one convolutional layer, 32 identical residual blocks, each composed of two convolutional layers, and one Relu layer in between, are used. The convolutional layers in the residual blocks have the same size as the feature maps and kernel size of first convolutional layer. In order to normalize the feature maps, a convolutional layer with batch normalization is applied after the residual blocks. A skip connection between the input of the first and the last residual block is used. Two more convolutional layers after the residual blocks are used. Finally, the last convolutional layer has one feature map which constructs the output frame.

Given $\mathcal{I} = \mathcal{P} \oplus \mathcal{C}$ as the concatenation of the prediction $\mathcal{P}$ and reconstruction $\mathcal{C}$ frames as input, producing the enhanced frame $\hat{\mathcal{O}}$, is formulated as

$$\hat{\mathcal{O}} = F_1^1(F_2^2(Bn(F_3^1(Res^{32}(F_2^1(\mathcal{I}))))) + \mathcal{I})), \tag{2}$$

where $F_1(.)$ and $F_2(.)$ are $3\times3\times256$ convolutional layer, with and without the Relu activation layer, respectively. Moreover, $F_3(.)$ is a $3 \times 3 \times 1$ convolutional layer with Relu activation layer. The superscript of each function indicates the number

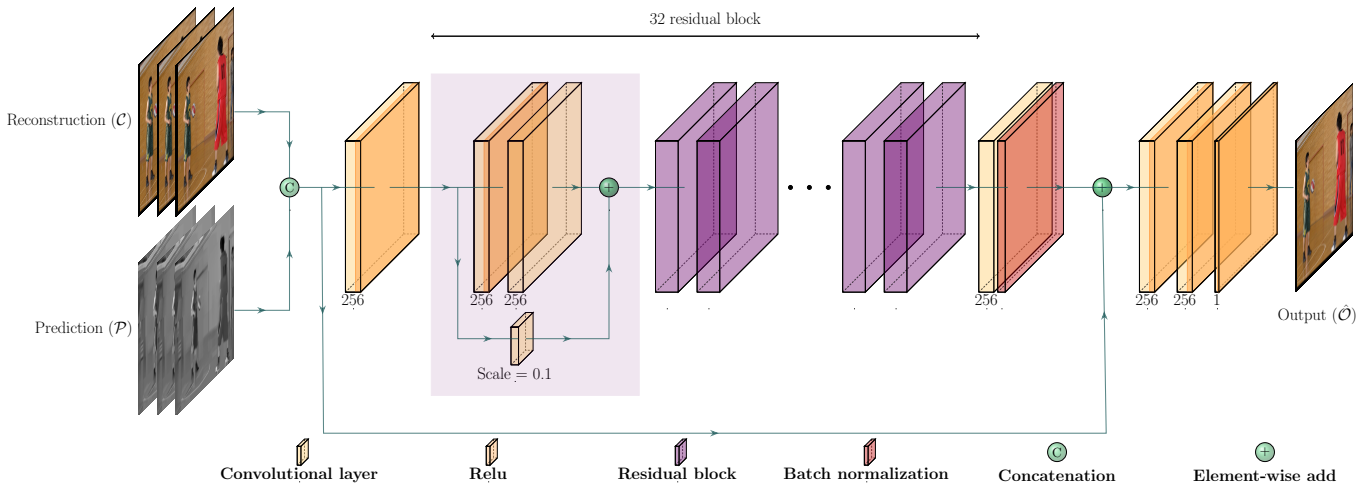

Fig. 3: Network architecture of the proposed method using the prediction and the reconstruction signal as the input.

of times they are repeated sequentially in the network architecture. Finally, Res and Bn are the residual block and batch normalization layer, respectively.

The task of the training phase is to optimize the parameters θ_{QE} of the above QE function, f_{QE}, expressed as

$$\hat{\mathcal{O}} = f_{QE}(\mathcal{C}, \mathcal{P}; \theta_{QE}). \tag{3}$$

The L_2 norm with respect to the original frame $\mathcal{O}$ is used as the cost function of the training phase

$$L_2(\mathcal{O} - \hat{\mathcal{O}}) = ||\mathcal{O} - \hat{\mathcal{O}}||_2^2. \tag{4}$$

In the proposed method, each color component of the decoded video (i.e. one luminance and two chrominance) is enhanced separately. For this purpose, one network for each component in different QPs is trained with the above network architecture and using the corresponding prediction signal of that component.

III. Experimental Results

As the proposed post-processing module is designed to enhance the quality of intra coded frames, two image datasets of DIV2K and Fliker2K are used for training. All images in the datasets are encoded in All-Intra (AI) configuration of the VVC Test Model version 5.0 (VTM-5.0) [1], using 6 QPs, between 22 and 47. The prediction information is extracted during decoding process for all datasets in all QP ranges.

The network was implemented and trained in pyTorch (1.4.0). For training, 64 $\times$ 64 patches of reconstruction and prediction frames were extracted randomly from the training dataset, with batch size of 32. The training started with the learning rate of 10^{-4} which was then decayed by the scale of 0.1 for each 100 epochs until 500 epochs. At the end of the training, a total of 3$\times$6 trained models were obtained for 3 components in 6 QPs.

In order to evaluate our method, the test sequences of JVET CTC (classes A1, A2, B, C, D, E) were encoded with the VTM-5.0 with each of the 6 QPs with the AI configuration. To study the effect of QE method in different bit-rates, two QP ranges were evaluated: 1) the CTC QP: (22, 27, 32, 37), and 2) high QP: (32, 37, 42, 47). The performance of different benchmark methods were measured using the Bjontegaard delta (BD) bit-rate saving metric based on the PSNR difference with respect to VTM-5.0 with no QE as an anchor.

Three state of the art VVC CNN-based QE methods are used as benchmark. First two methods are JVET contributions [4], [14] proposing QE methods as post processing. Both of these methods deploy a slightly simpler network architecture with the QP map and use the reconstruction signal as the only input to the network. To assess the benefit of using IPM as input to the network, we also present the results for the proposed method with only reconstruction frame as input (denoted "proposed - without prediction")

Table I presents the performance of our proposed method against the anchor compared with the two benchmark methods. It can be seen that in the CTC QP range, the proposed method can achieve an average BD-rate gain of 6.7%, 12.6% and 14.5% on Y, U and V components, respectively. In the same QP range, it is also observed that the proposed method with the prediction signal outperforms the proposed method without the prediction signal by 0.9%, 8.1% and 4.8%, on Y, U and V components, respectively. Compared to the other two JVET solutions, the proposed method shows a significant gain, in the CTC QP range.

At high QP range, where artifacts are significantly stronger, the only comparison is between the proposed method with and without the prediction signal. As can be seen in Table I, the proposed method can achieve an average BD-rate gain of 8.3%, 15.8% and 16.2% on Y, U and V components, respectively. Same as CTC QP range, the use of the prediction signal in high QPs also further increases the gain with an average BD-rate of 1.3%, 7.1% and 3.5% on Y, U and V components, respectively.

In both QP ranges, the achieved BD-rate gain of using the prediction signal is relatively higher for the U and V components than for the Y component. This can be due to

TABLE I: Performance comparison in percentage (%) of the proposed method against the VVC in terms of BD-Rate.

Class	Sequence	CTC QP (22-37)												High QP (32-47)					
		JVET-N0254 [4] (VTM 4.0)			JVET-N0169 [10] (VTM 4.0)			Proposed (VTM 5.0)						Proposed (VTM 5.0)					
								Without prediction			With Prediction			Without prediction			With Prediction		
		Y	U	V	Y	U	V	Y	U	V	Y	U	V	Y	U	V	Y	U	V
A1	Tango	-0.9	-2.7	-3.3	-3.7	-7.8	-8.1	-4.3	-4.4	-9.1	-5.4	-21.8	-21.1	-6.4	-10.5	-11.9	-7.9	-20.2	-16.9
	FoodMarket	-1.3	-1.7	-2.3	-3.8	-3.8	-4.1	-9.9	-3.8	-9.7	-11.3	-12.2	-13.3	-7.8	-8.9	-10.9	-8.8	-12.8	-11.0
	CampFire	-0.6	-0.6	-3.1	-2.5	-9.5	-8.3	-3.3	-3.2	-11.1	-4.0	-8.2	-20.2	-6.4	-8.3	-11.2	-7.6	-10.0	-18.8
	Average	**-1.0**	**-1.6**	**-2.9**	**-3.3**	**-7.0**	**-6.8**	**-5.8**	**-3.8**	**-10.0**	**-6.9**	**-14.1**	**-18.2**	**-6.9**	**-9.3**	**-11.3**	**-8.1**	**-14.4**	**-15.6**
A2	CatRobot	-2.2	-3.7	-3.8	-4.6	-7.3	-10.8	-5.7	-4.7	-11.3	-6.4	-14.0	-18.8	-6.7	-13.1	-14.4	-8.0	-18.0	-17.7
	Daylight	-1.1	-4.3	-1.2	-3.5	-4.6	-5.9	-2.7	-1.7	-5.3	-3.7	-11.9	-11.5	-7.4	-7.9	-6.8	-8.9	-11.9	-9.2
	ParkRunning	-1.1	-0.2	-0.4	-3.6	-3.7	-3.4	-3.8	-0.6	-1.6	-4.4	-4.5	-4.4	-4.6	-2.2	-2.3	-5.6	-5.6	-5.3
	Average	**-1.5**	**-2.7**	**-1.8**	**-3.9**	**-5.2**	**-6.7**	**-4.1**	**-2.3**	**-6.1**	**-4.8**	**-10.1**	**-11.6**	**-6.2**	**-7.8**	**-7.8**	**-7.5**	**-11.8**	**-10.7**
B	MarketPlace	-0.9	-2.6	-2.9	-3.6	-3.7	-3.6	-4.2	-3.3	-7.9	-5.2	-13.6	-14.1	-5.2	-10.1	-11.3	-6.5	-17.6	-15.9
	RitualDance	-2.2	-2.3	-4.5	-6.0	-3.7	-4.5	-8.9	-4.3	-11.4	-10.2	-14.2	-16.9	-7.4	-9.7	-13.8	-8.9	-17.0	-17.4
	Cactus	-0.7	-2.5	-0.8	-4.0	-4.2	-9.1	-4.2	-2.1	-10.9	-4.9	-10.2	-15.7	-6.9	-9.9	-15.2	-8.0	-15.3	-18.0
	BasketballDrive	-0.3	-2.3	-3.5	-4.2	-11.8	-14.7	-5.1	-8.2	-16.1	-6.2	-18.9	-21.0	-6.3	-12.8	-17.4	-7.7	-17.9	-19.4
	BQTerrace	-0.5	-1.2	-3.7	-2.3	-7.2	-6.7	-3.4	-4.3	-8.2	-3.9	-13.6	-12.5	-7.5	-9.9	-9.4	-8.7	-15.3	-13.7
	Average	**-0.9**	**-2.2**	**-3.1**	**-4.0**	**-6.1**	**-7.7**	**-5.2**	**-4.5**	**-10.9**	**-6.1**	**-14.1**	**-16.0**	**-6.7**	**-10.5**	**-13.4**	**-8.0**	**-16.6**	**-16.9**
C	BasketballDrill	-3.0	-3.2	-5.2	-8.1	-14.8	-20.5	-9.1	-11.8	-24.3	-10.3	-23.3	-28.0	-8.7	-19.2	-21.2	-10.2	-21.2	-23.0
	BQMall	-2.1	-2.5	-4.3	-6.3	-5.9	-7.1	-6.7	-4.5	-7.6	-7.4	-11.7	-11.8	-7.6	-10.4	-11.3	-8.7	-16.9	-15.9
	PartyScene	-1.8	-1.3	-1.5	-4.2	-4.5	-4.7	-4.4	-3.1	-6.0	-4.8	-8.5	-10.0	-6.6	-9.1	-9.7	-7.5	-14.8	-15.3
	RaceHorses	-0.7	-2.4	-2.4	-3.6	-5.9	-9.3	-3.3	-3.6	-9.2	-3.8	-8.1	-12.3	-4.8	-14.0	-18.4	-5.7	-20.4	-22.8
	Average	**-1.9**	**-2.3**	**-3.3**	**-5.6**	**-7.8**	**-10.4**	**-5.9**	**-5.7**	**-11.8**	**-6.6**	**-12.9**	**-15.5**	**-6.9**	**-13.2**	**-15.2**	**-8.0**	**-18.3**	**-19.2**
D	BasketballPass	-2.4	-1.0	-3.5	-7.1	-9.2	-13.5	-8.0	-8.7	-15.7	-8.8	-18.5	-18.0	-8.7	-14.5	-18.4	-9.8	-19.6	-22.1
	BQSquare	-2.0	0.2	-3.7	-5.3	-2.0	-6.0	-6.4	-0.5	-5.3	-6.8	-4.6	-9.5	-8.9	-3.0	-14.5	-9.9	-6.1	-17.1
	BlowingBubble	-2.0	-0.5	-2.8	-4.9	-5.9	-5.2	-5.3	-4.2	-6.6	-5.9	-11.5	-11.2	-6.3	-9.2	-9.8	-7.4	-14.8	-13.6
	RaceHorses	-2.5	-2.4	-3.5	-6.1	-8.7	-12.1	-5.9	-6.1	-11.7	-6.4	-12.8	-15.3	-6.6	-11.9	-13.7	-7.6	-17.4	-17.4
	Average	**-2.2**	**-0.9**	**-3.4**	**-5.8**	**-6.5**	**-9.2**	**-6.4**	**-4.9**	**-9.8**	**-7.0**	**-11.9**	**-13.5**	**-7.6**	**-9.6**	**-14.1**	**-8.7**	**-14.5**	**-17.5**
E	FourPeople	-3.1	-1.6	-1.6	-7.2	-5.5	-5.6	-8.3	-4.1	-5.9	-9.3	-9.6	-9.5	-8.0	-10.6	-11.3	-9.6	-14.9	-14.5
	Johnny	-2.0	-1.7	-3.1	-6.3	-9.4	-8.7	-8.2	-7.0	-10.5	-9.4	-15.0	-13.1	-8.3	-15.6	-15.0	-9.5	-20.3	-16.7
	KristenAndSara	-2.6	-1.5	-1.7	-6.4	-8.0	-7.3	-7.4	-4.6	-8.6	-8.2	-11.3	-12.0	-7.8	-14.7	-12.6	-9.0	-19.5	-15.3
	Average	**-2.6**	**-1.6**	**-2.1**	**-6.6**	**-7.7**	**-7.2**	**-7.9**	**-5.2**	**-8.3**	**-8.9**	**-12.0**	**-11.5**	**-8.0**	**-13.6**	**-13.0**	**-9.4**	**-18.2**	**-15.5**
	All	**-1.6**	**-1.9**	**-2.9**	**-4.9**	**-6.7**	**-8.2**	**-5.8**	**-4.5**	**-9.7**	**-6.7**	**-12.6**	**-14.5**	**-7.0**	**-10.7**	**-12.7**	**-8.3**	**-15.8**	**-16.2**

the fact that in VVC, there are advanced tools for chroma coding to exploit the redundancies. Examples of such tools are Luma Mapping with Chroma Scaling (LMCS), Joint Cb-Cr residual coding (JCCR), Cross-Component Linear Modeling (CCLM) and a specific chroma IPM called luma Derived Mode (DM) [1]. The use of coding information such as intra prediction might enable the CNN-based QE to benefit from the existing correlations and more efficiently predict the compression artifacts.

IV. Conclusion

In this paper, a CNN-based quality enhancement method was proposed for VVC coded frames, that benefits from the coding information in the intra prediction signal of each frame. The experiments showed that using prediction information can significantly improve the performance of the CNN-based enhancement methods, both for luma and chroma components of intra frames. The best explanation for the observed improvements is that exposing the CNN training process to coding information of the sequences, along with their ground-truth original signal, helps them is learning the pattern of compression artifacts. Hence, when the networks are used for the QE task of actual compressed sequences, they can more efficiently recover the lost information.

References

[1] B. Bross et al. Versatile video coding (draft 7). In *JVET-P2001, Geneva, Switzerland*, October 2019.

[2] D. Liu et al. Deep learning-based video coding: A review and a case study. *ACM Computing Surveys (CSUR)*, 53(1):1–35, 2020.

[3] S. Ma et al. Image and video compression with neural networks: A review. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[4] Y. Wang et al. CE13: Dense residual convolutional neural network based in-loop filter. In *JVET-N0254, Geneva, Switzerland*, March 2019.

[5] S. Wan et al. Integrated in-loop filter based on CNN. In *JVET-O0079, Gothenburg, Sweden*, July 2019.

[6] F. Bossen et al. JVET common test conditions and software reference configurations for sdr video. In *JVET-M1001, Marrakesh, Morocco*, January 2019.

[7] Y. Dai et al. Ce13: Experimental results of CNN-based in-loop filter (ustc). In *JVET-N0513, Geneva, Switzerland*, Mar. 2019.

[8] K. Kawamura et al. Evaluation results of CNN based in-loop filtering. In *JVET-N0710, Geneva, Switzerland*, Mar. 2019.

[9] YH. Lam et al. Efficient adaptation of neural network filter for video compression. *arXiv preprint arXiv:2007.14267*, 2020.

[10] X. Meng et al. Enhancing quality for VVC compressed videos by jointly exploiting spatial details and temporal structure. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1193–1197. IEEE, 2019.

[11] D. Ma et al. MFRNet: a new CNN architecture for post-processing and in-loop filtering. *arXiv preprint arXiv:2007.07099*, 2020.

[12] W. Wang et al. Attention-based dual-scale CNN in-loop filter for versatile video coding. *IEEE Access*, 7:145214–145226, 2019.

[13] F. Zhang et al. Enhancing vvc through CNN-based post-processing. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2020.

[14] J. Yao et al. Convolutional neural network filter (CNNF) for intra frame. In *JVET-N0169, Geneva, Switzerland*, March 2019.

[15] F. Nasiri et al. A study on the impact of training data in CNN-based super-resolution for low bitrate end-to-end video coding. *IEEE International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–5, 2020.

[16] L. Bee et al. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017.