NEURAL NETWORK-BASED ERROR CONCEALMENT FOR VVC

Martin Benjak, Yasser Samayoa, Jörn Ostermann

Institut für Informationsverarbeitung
Gottfried Wilhelm Leibniz Universität Hannover
30167 Hannover, Germany
Email: {benjak, samayoa, office}@tnt.uni-hannover.de

ABSTRACT

In this paper we introduce an error concealment method for VVC based on deep recurrent neural networks, which employs the PredNet model to estimate missing video frames by using past decoded frames. The network is trained using the BVI-DVC data set to infer even full-HD frames. We integrated our proposed model in the VVC reference software VTM for its evaluation. It performs, in average, 6 dB or up to 5 dB better than the frame copy model in terms of PSNR measurements for a concealed I-frame or P-frame, respectively.

***Index Terms*—** VVC, video communication, video coding, error concealment

1. INTRODUCTION

The evolution of technologies for the display and recording of video signals has been responding to the rising demand for higher resolution devices. To meet these demands, the state-of-the-art video coding standard Versatile Video Coding (VVC) [1] has been released in summer 2020. This is driving all types of communication systems to increase their capacity of conveying information, from which video contents are predominant. For instance, by 2022 nearly four-fifths of the world's mobile data traffic will be video [2]. Many applications like video surveillance, tele-medicine and smart car navigation systems require greater resolution and lower latency video communication systems [3].

For the transmission and storage of video signals, the imperative systems are video coding, channel coding and communication systems. For low-delay applications however, video transmission imposes extra challenges because error-free output can not be guaranteed at the decoder side by any means. This forces the execution of error concealment (EC) algorithms in the video decoder to minimize the impact of errors that cannot be corrected by the channel decoder. It is worth noting that the impact of an uncorrected error increases with the coding efficiency. On the one hand, each video compression standard reaches a higher coding efficiency in comparison to its predecessors. On the other hand, the complexity of a suitable EC increases as well. Additionally, in the last two video coding standards, VVC and High Efficiency Video Coding (HEVC) [4], error resilience mechanisms have not been included and there is no suggestion for EC. These new standards assume error-free transmissions, which can not be guaranteed for real systems.

The problem of EC has been of great importance since the beginning of digital video communication systems. Several solutions have been proposed for standards prior to HEVC [5, 6, 7, 8]. These algorithms were developed for video codecs based on macroblocks (MB), in which the spatio-temporal correlation of MBs is exploited to conceal lost MBs. HEVC and VVC abandoned the MB-based coding scheme and therefore require new EC solutions. Few EC algorithms for HEVC can be found in the literature [9, 10, 11, 12, 8]. These schemes address EC with analytical methods by exploiting spatio-temporal information available in the decoder to construct the lost portion of the video. In [13], a deep neuronal network was trained to emulate EC for a single lost slice assuming a frame is divided in multiple slices. Its performance was not measured within any coding standard. Until now, there are no publications regarding EC for VVC.

In this paper we propose a machine learning-based EC algorithm for VVC. We focus on low-latency applications for video communication systems over error prone channels. One slice per frame is assumed, such that just one erroneous bit in the encoded bit-stream can completely corrupt a whole video frame and produce the worst video quality degradation for subsequent frames due to inter-prediction. Our model makes use of a deep recurrent neural network (RNN) to generate an estimated version of the lost frame from previously decoded frames. The impact of the concealed frame on the video quality is evaluated with the reference software VVC Test Model (VTM). Currently, VTM has no capability to detect and conceal a lost slice, which means that our proposed EC algorithm is implemented and adapted to the VTM decoder.

The remainder of this paper is organized as follows. In Section 2, a high-level introduction of VVC is given. In Section 3, we present the proposed algorithm. In Section 4, an evaluation and experimental results are given and Section 5 provides a conclusion for this paper.

2. VERSATILE VIDEO CODING (VVC)

Initiated in 2018, the Joint Video Expert Team (JVET), a joint collaborative team established by the ITU-T Video Coding Expert Group (VCEG) and the ISO/IEC Moving Experts Group (MPEG) organizations, finalized the standardization of VVC and released its final draft in 2020 [1]. One of the main goals of VVC was to increase the compression capability in comparison to its predecessors. It provides a BD-rate gain of 30% compared to HEVC [14].

The basic coding structure of VCC is the same as its predecessors since H.261: block-based hybrid video coding. This hybrid scheme combines intra and inter (motion compensated) prediction, transform coding techniques, quantization, entropy coding and loop filtering. Intra coding relies on previously coded parts of the current picture to predict a new block within this picture. Inter coding additionally utilizes temporal redundancy between consecutive pictures to improve the prediction. Conceptually, previously reconstructed pictures are stored in a reference picture buffer and are used to make a prediction for the currently coded block via motion compensated prediction. VVC, however, enhanced legacy coding tools and introduced new ones compared to HEVC [1, 15]. Among others these improvements are: more flexible block partitioning structures with up to 128×128 luma samples; quad-tree and multiple-type tree partition strategy; sub-block-based motion compensation; adaptive loop filter for post filtering; 67 intra prediction modes (32 more than HEVC); 1/16-pel luma and 1/32-pel chroma motion vector accuracy; different transforms and quantizers; screen content coding support; affine motion compensation and bi-directional optical flow. In consequence, VVC achieves a better prediction accuracy than HEVC. However, motion compensation still contributes the most to its coding gain, which also increases its sensibility to errors.

3. PROPOSED ERROR CONCEALMENT METHOD

3.1. System Description

On the transmitter side, the VTM video encoder compresses the input video and delivers a bitstream or Network Abstraction Layer (NAL) unit stream to the channel encoder and communication system blocks. The channel encoder intelligently adds redundancy to the bitstream to increase its robustness against errors. These encoded bits are conveyed over an error prone channel. On the receiver side, the forward error correction (FEC) block recovers the bitstream from the encoded bits by removing the redundancy added at the transmitter side while it detects and corrects the errors added by the channel. Afterwards, the bitstream is passed to the VTM video decoder. Figure 1 shows a simplified block diagram of our EC solution integrated to the VTM video decoder. The CABAC block maps the received bitstream into syntax elements which after the inverse transform and quantizer are added to the prediction values. The prediction block contains the inter and intra prediction and the buffer block holds the reference frames needed for the inter prediction. For frames without errors, the switch is in the decoded frame (df) position.

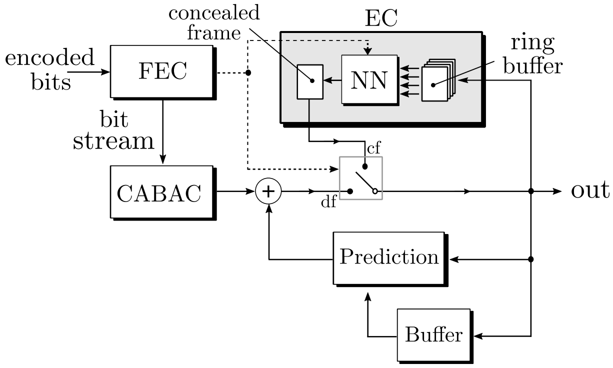

Fig. 1: Block diagram of VVC and NN method

FEC triggers the EC algorithm when an error is detected in the bit stream (dashed line). In this paper, we configure a slice to contain an entire frame which makes sense for low-delay applications [16]. Therefore, a NAL unit contains a whole frame as well. Just one erroneous bit in a NAL unit is enough to prevent CABAC from recovering the syntax elements of an entire frame and thus the frame is considered to be lost. If an error is detected, the EC algorithm is started: The NN estimates the lost frame from the L previous frames contained already in the ring buffer. Also, the switch is changed to the concealed frame (cf) position, indicating that the lost frame will be replaced with the concealed frame in the output sequence and saved in the buffer block for the inter prediction of the following frames. Note that the ring buffer can contain both correctly decoded and concealed frames.

3.2. Neural Network-Based Frame Estimation

In this paper a neural network-based frame estimation, in short NN, is proposed as an EC algorithm. Our model employs the RNN architecture from the PredNet model [17] to estimate corrupted and thus missing frames. This network consists of four stacked modules, each trying to make predictions for its input. The prediction is generated by a convolutional layer from a recurrent representation. The difference between estimated and actual frame (estimation error) is passed through a convolutional layer and given as the input to the next layer. In opposite order, the recurrent representation of each module is generated using a Long Short-Term Memory (LSTM) layer with the estimated error of the last time step, the recurrent representation of the last layer and the recurrent representation of the last time step. This process is repeated with the next frame of a sequence in each time step.

After the last time step, the concealed frame in the sequence is obtained by an arbitrary input frame into the first module.

The model was trained using the BVI-DVC data set [18] which contains 800 sequences with 64 frames each. The spatial resolution varies between 3840x2176 and 480x272. Due to GPU-memory limitations, the model was trained using a resolution of 480x272. To overcome this limitation and still enable the model to infer full-HD sequences, the data set was preprocessed to ensure that the model learns scale-invariant features. The sequences with a resolution of 3840x2176 were down-scaled to 1920x1088 and afterwards the following preprocessing was performed: (a) All sequences were downscale to 480x272. (b) All sequences were split into non-overlapping 480x272 parts. (c) A central 480x272 section was cropped from all sequences. After this procedure, which also serves as a form of data augmentation, our training data set contained 8600 sequences with 64 frames each. Following [17], the layer channel size was set to $\{3, 48, 96, 192\}$. We trained the model over 2 epochs with the full training data set using Adam as optimizer, $\beta_1 = 0.1$, $\beta_2 = 0.999$ and an initial learning rate of 0.0001. After 1 epoch, we linearly decreased the learning rate down to 0.00008 at the end of the training.

4. SIMULATION RESULTS AND DISCUSSION

4.1. System Configuration

All simulations were performed according to the JVET common test conditions (CTC) for neural network-based video coding [19]. Sequences of the classes B, C, D and E were encoded by means of the unmodified VTM 10.0 with quantization parameter values $\text{QP} = \{22, 27, 32, 37, 42\}$. However, for the figures presented in this paper we use $\text{QP} = 22$. Class A was not included in the simulations due to GPU memory limitations and class F was not included since screen content is not present in the BVI-DVC data set. We modified the low-delay configuration such that a set of 8 frames forms a group of pictures (GOP) with an I-slice as its first frame and the rest being P-slices. One slice per frame is selected. The VTM 10.0 decoder is extended with EC capabilities as described in Section 3, i.e., the well-known frame copy (FC) and our NN methods are integrated. FC just conceals a lost frame by copying the previous decoded frame while NN estimates it utilizing the $L = 5$ previously decoded frames. It should be noted that the input and output for NN are whole frames and not patches. We also implemented an optical flow [20] EC method, but the PSNR gain over FC was only 0.38 dB for QP 22 while being computationally far more expensive. Thus we did not include this method into our evaluation.

4.2. Results

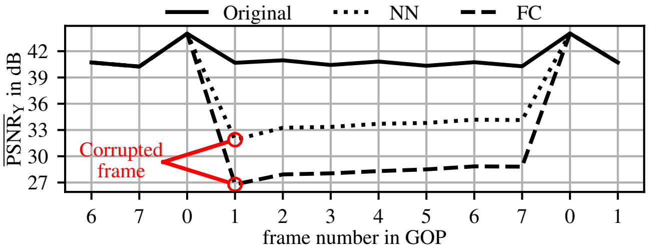

Fig. 2: $\overline{\text{PSNR}}_\text{Y}$ over a relative frame number in a GOP. The second frame of the second GOP is corrupted which is 8 frames long and has a structure of IPPPPPPP.

In this section, we evaluate the performance of NN and compare it with FC. We first present the average PSNR for each frame position within a GOP, as shown in Figure 2. Each video of the test set was divided into segments of three consecutive GOPs. In each of these segments, an error in the second frame of the second GOP was introduced such that the frame is lost. Each EC method produces an estimated frame to replace the lost one. PSNR_Y is computed for each frame, then it is averaged over all frames belonging to the same relative frame position in every video segment of three GOPs resulting in $\overline{\text{PSNR}}_\text{Y}$ in Figure 2. As it is expected, a frame lost jeopardizes the video quality significantly until the arrival of the next intra frame. As it can be observed, once the information has been lost, it can hardly be recovered using the information of the following frames withing the GOP. For this reason, it

Table 1: PSNR values for luma and chroma components averaged over all GOPs of the videos in a class and different QPs when the second frame in every GOP is corrupted.

Video class	QP	Original			NN			FC		
		Y	Cb	Cr	Y	Cb	Cr	Y	Cb	Cr
B	22	40.8	43.9	45.7	33.0	40.6	40.9	27.5	38.7	38.8
	27	38.1	42.5	43.9	32.2	40.0	40.4	27.1	38.5	38.6
	32	36.0	41.3	42.4	31.5	39.3	39.7	27.0	38.2	38.2
	37	33.8	39.6	40.7	30.5	38.3	38.7	26.5	37.7	37.7
	42	31.5	38.6	39.3	29.1	37.5	37.8	25.8	37.2	37.1
C	22	40.4	42.9	44.2	32.5	39.1	39.5	26.6	36.2	36.1
	27	37.0	40.6	41.6	31.7	38.0	38.4	26.3	35.7	35.7
	32	34.0	38.8	39.6	30.5	36.9	37.3	25.7	35.0	35.1
	37	31.3	37.1	37.8	28.9	35.7	36.0	24.9	34.3	34.4
	42	28.7	35.6	36.0	27.0	34.5	34.7	24.0	33.5	33.6
D	22	39.8	42.9	43.5	33.7	40.2	40.1	29.4	38.9	38.5
	27	35.8	40.4	40.8	32.3	38.8	38.8	29.0	37.7	37.4
	32	32.6	38.5	38.7	30.4	37.3	37.3	27.9	36.5	36.0
	37	29.8	36.7	36.8	28.4	35.9	35.7	26.5	35.3	34.8
	42	27.2	35.4	35.2	26.2	34.8	34.4	24.9	34.4	33.9
E	22	43.3	47.9	48.9	40.8	46.4	47.3	38.5	46.3	47.0
	27	41.6	46.4	47.4	39.7	45.3	46.2	38.0	45.2	46.0
	32	39.5	44.7	45.5	38.2	44.0	44.7	37.0	43.9	44.6
	37	37.0	42.5	43.6	36.1	42.0	43.1	35.4	42.0	43.1
	42	34.2	41.1	42.1	33.7	40.8	41.8	33.4	40.7	41.8

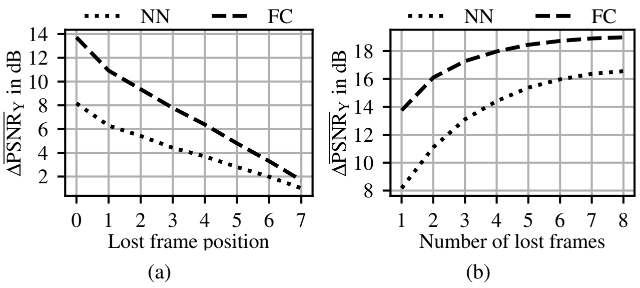

Fig. 3: Quality measurement $\Delta\overline{\text{PSNR}}_{\text{Y}}$ between error-free and concealed video plotted over the relative position of a lost frame in a GOP (a) and the number of consecutive lost frames within a GOP starting with the I-frame (b).

is essential to estimate a lost frame as good as possible. As shown in the figure, NN gives the highest $\overline{\text{PSNR}}_{\text{Y}}$ compared to the FC method. Table 1 confirms this same tendency in more detail. It gives an average PSNR over all frames in a GOP and over all GOPs in every video of the CTC. In this setting, the second frame of every GOP is lost. FC and NN estimate the lost frames assuming that the previous frames are error-free. It can been seen, that NN performs better than FC for all classes. The difference in performance between NN and FC is more accentuated for videos with high motion content, however, in class E their differences are less pronounced. Class E contains video-conference settings, i.e., little movement with static camera and static background. In this class, the background covers a considerable area of a frame, thus, a static background would be better concealed by FC than by NN because NN introduces additional noise inevitably produced by its own network.

In Figure 3 the difference between frames of the error-free and the error-concealed videos is measured and averaged over an i-th GOP, i.e., $\Delta\text{PSNR}_{\text{Y},i} = \text{mean}\{\text{PSNR}_n - \text{PSNR}_{\text{EC},n}\}$ for all n, where PSNR and PSNR_{EC} are the error-free and concealed PSNR measurement of the n-th frame respectively, $0 \leq n \leq 7$. The average over all i in all videos is denoted by $\Delta\overline{\text{PSNR}}_{\text{Y}}$ which is plotted in Figure 3 over (a) the relative position of a lost frame in a GOP and (b) the number of consecutive lost frames starting at $n = 0$. Both NN and FC assume that the previous GOP is error-free. In (a) the maximum $\Delta\overline{\text{PSNR}}_{\text{Y}}$ is found when the intra-frame is lost and monotonically decreases with the corrupted frame position. Clearly, the largest loss of quality is due to the interdependence of frames caused by inter coding. The more frames depend on a concealed frame, the higher $\Delta\overline{\text{PSNR}}_{\text{Y}}$ is. In absence of frames interdependence, $\Delta\overline{\text{PSNR}}_{\text{Y}} \approx 1$ dB for both EC, e.g., when $n = 7$. Moreover, in case of more than one consecutive corrupted frame withing a GOP, the loss in PSNR increases as shown in (b). The loss increases rapidly and after a few consecutive lost frames it may no longer be worth to conceal them anymore.

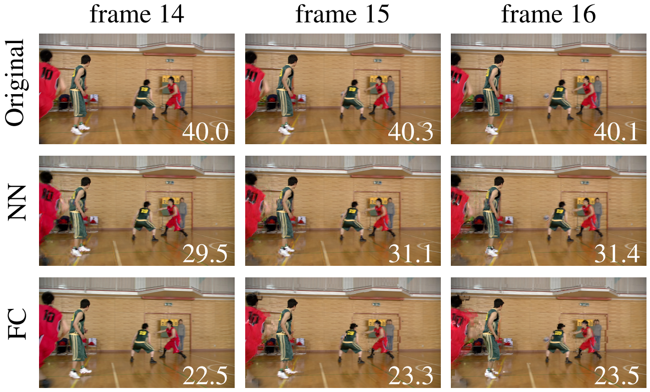

Fig. 4: The top row shows the original, thus, error-free frames of the BasketballDrive sequence. The 14th frame is estimated using NN and FC approach, middle and bottom row respectively. In each frame its corresponding PSNR_{Y} is given.

Figure 4 shows a visual comparison of the EC methods in which the 14th frame is concealed. The uncorrupted and, therefore, original frame is presented as well. Frames 15 and 16 are also presented to help visualize the inter-frame error propagation. The differences are clearer by zooming in on the figure. As it can be seen, the difference between the original and the concealed frame is up to 10.5 dB and 17.5 dB for NN and FC respectively. In concordance with Figure 2, these differences do not changes considerably on the following frames. Interesting are the subjective comparisons of player No. 10 and the frame of the door. As it can be noticed, the NN method is capable of predicting the movement not only of the camera but also of the players. $\text{QP} = 22$ was used for encoding parameters of Figures 2, 3 and 4, however, we obtained similar results with $\text{QP} = \{27, 32, 37, 42\}$.

5. CONCLUSION

This paper presents a neural network-based error concealment algorithm for VVC by estimating a lost frame from five consecutive past frames. Its performance was evaluated for low-delay applications of video communication systems over error prone channels using the CTC for neural network-based video coding. We implemented both NN and FC methods in the reference software VTM 10.0. NN gives the highest PSNR in comparison to FC for all classes, e.g., NN performs for an estimated frame on average 5 dB better than FC and if the interdependency is considered up to 6 dB. We found that the differences between performance is more accentuated in sequences with high amount of motions. By means of a visual comparison it was shown that NN method is capable of predicting the movement not only of the camera but also of the content in a video. A well trained neural network can be used to estimate a lost frame at the decoder even for full-HD resolution videos, which makes it a viable option as an error concealment solution for VVC.

6. REFERENCES

[1] B. Bross, J. Chen, S. Liu, and Y. Wang, "Versatile video coding (draft 10)," *ITU-T and ISO/IEC JVET-S2001*, 2020.

[2] GMDT Forecast, "Cisco visual networking index: global mobile data traffic forecast update, 2017–2022," *Update*, vol. 2017, pp. 2022, 2019.

[3] Cisco, "Cisco annual internet report (2018–2023) white paper," 2020.

[4] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[5] Y. Wang and Q. Zhu, "Error control and concealment for video communication: a review," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974–997, 1998.

[6] J. Suh and Y. Ho, "Error concealment based on directional interpolation," *IEEE Transactions on Consumer Electronics*, vol. 43, no. 3, pp. 295–302, 1997.

[7] W. M. Lam, A. R. Reibman, and B. Liu, "Recovery of lost or erroneously received motion vectors," in *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1993, vol. 5, pp. 417–420 vol.5.

[8] M. Usman, X. He, M. Xu, and K. M. Lam, "Survey of error concealment techniques: Research directions and open issues," in *2015 Picture Coding Symposium (PCS)*, May 2015, pp. 233–238.

[9] Chang Liu, Ran Ma, and Zhaoyang Zhang, "Error concealment for whole frame loss in hevc," in *Advances on Digital Television and Wireless Multimedia Communications*, Wenjun Zhang, Xiaokang Yang, Zhixiang Xu, Ping An, Qizhen Liu, and Yue Lu, Eds., Berlin, Heidelberg, 2012, pp. 271–277, Springer Berlin Heidelberg.

[10] Y. Chang, Y. A. Reznik, Z. Chen, and P. C. Cosman, "Motion compensated error concealment for hevc based on block-merging and residual energy," in *2013 20th International Packet Video Workshop*, Dec 2013, pp. 1–6.

[11] T. Lin, N. Yang, R. Syu, C. Liao, and W. Tsai, "Error concealment algorithm for hevc coded video using block partition decisions," in *2013 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC 2013)*, Aug 2013, pp. 1–5.

[12] Y. Zhang and Z. Li, "Multi-hypothesis-based error concealment for whole frame loss in hevc," in *MultiMedia Modeling*, Cham, 2018, pp. 342–354, Springer International Publishing.

[13] A. Sankisa, A. Punjabi, and A. K. Katsaggelos, "Video error concealment using deep neural networks," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct 2018, pp. 380–384.

[14] T. Laude, Y. Adhisantoso, J. Voges, M. Munderloh, and J. Ostermann, "A comprehensive video codec comparison," *APSIPA Transactions on Signal and Information Processing*, vol. 8, 2019.

[15] X. Xu and S. Liu, "Recent advances in video coding beyond the hevc standard," *APSIPA Transactions on Signal and Information Processing*, vol. 8, 2019.

[16] Y. Samayoa and J. Ostermann, "Parameter selection for a video communication system based on hevc and channel coding," in *2020 IEEE Latin-American Conference on Communications (LATINCOM)*, Nov 2020, pp. 1–5.

[17] W. Lotter, G. Kreiman, and D. Cox, "Deep Predictive Coding Networks for Video Prediction and Unsupervised Learning," in *International Conference on Learning Representations*, 2017.

[18] D. Ma, F. Zhang, and D. Bull, "Bvi-dvc: A training database for deep video compression," *arXiv:2003.13552*, 2020.

[19] S. Liu, A. Segall, E. Alshina, and R. Liao, "Jvet common test conditions and evaluation procedures for neural network-based video coding technology," *ITU-T and ISO/IEC JVET-T2006*, 2020.

[20] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Scandinavian conference on Image analysis*. Springer, 2003, pp. 363–370.