Enhanced Video Compression based on Effective Bit depth Adaptation

Fan Zhang, Mariana Afonso and David R. Bull

Abstract

This paper presents a novel Convolutional Neural Network (CNN) based effective bit depth adaptation approach (EBDA-CNN) for video compression. It applies effective bit depth down-sampling before encoding and reconstructs the original bit depth using a deep CNN based up-sampling method at the decoder. The proposed approach has been integrated with the High Efficiency Video Coding reference software HM 16.20, and evaluated under the Joint Video Exploration Team Common Test Conditions using the Random Access configuration. The results show consistent coding gains on all tested sequences, with an average bitrate saving of 6.4%, based on Bjøntegaard Delta measurements using PSNR.

Index Terms

Effective bit depth adaptation, EBDA-CNN, video compression, machine learning based compression, HEVC

I. Introduction

The increased requirement for higher quality, more immersive video content, with higher frame rate, greater spatial resolution and higher dynamic range (bit depth), is a primary driver for Internet streaming. CISCO predict that, by 2022, there will be 4.8ZB of global Internet traffic per year with 82% being video [1]. In this context, how we represent and communicate video (via compression) is key in ensuring that the content is delivered at an appropriate quality, while maintaining compatibility with the transmission bandwidth.

To address this, ISO/IEC MPEG and ITU have recently initiated the development of a new coding standard, Versatile Video Coding (VVC) [2, 3], targeted at increasing coding gain by 30-50% compared to High Efficiency Video Coding (HEVC) [4]. At the same time, the Alliance for Open Media (AOM) has been founded to develop open-source and royalty-free codecs, such as AOMedia Video 1 (AV1), which can compete with MPEG standards. Although the development of these codecs are still ongoing, they are still anticipated to adopt a similar compression architecture to that used in existing video codecs, such as HEVC and H.264/Advanced Video Coding (AVC) [5], but with more sophisticated coding tools.

On the other hand, a number of approaches have been proposed to improve coding efficiency through resolution (spatial or temporal) re-sampling, which encode a video at a lower resolution and reconstruct the full resolution at the decoder. Many of these methods apply spatial resolution adaptation at various compression levels or as pre-/post-processing steps [6, 7], and in most cases re-sampling is only applied for relatively low bitrate applications. Temporal resolution (frame rate) adaptation has also been employed to improve video compression efficiency [8, 9], and in these methods, an optimal frame rate is selected based on various frame rate dependent quality assessment methods. It should be noted that most re-sampling methods focus solely on spatial or temporal resolution, with very little research reported on the impact of bit depth re-sampling.

Thanks to breakthroughs in Artificial Intelligence (AI) technology, machine learning is becoming an important tool for video compression. It however remains an underdeveloped research area, with most existing work focused on the improvement of conventional coding tools, such as intra prediction [10, 11], inter prediction [12, 13] and in-loop filters [14, 15]. Some work has been reported on the use of deep learning based resolution adaptation methods [6, 16, 17], which have demonstrated resolution adaptation across a wider bitrate range and offer significant coding gains.

Inspired by our previous work [16], a novel Convolutional Neural Network (CNN) based effective bit depth adaptation approach (EBDA-CNN) is proposed for video compression, which reduces the effective bit depth of an

The authors acknowledge funding from EPSRC (EP/M000885/1) and the NVIDIA GPU Seeding Grants. The authors are with the Department of Electrical and Electronic Engineering, University of Bristol, Bristol, BS8 1UB, UK. {fan.zhang, mariana.afonso, dave.bull}@bristol.ac.uk

input video before encoding, and reconstructs the original bit depth at the decoder. The data precision and internal bit depth during encoding and decoding remain the same. In order to reconstruct high quality full bit depth content at the decoder, a modified CNN was employed, trained on HEVC compressed content. This approach has been integrated with an HEVC reference codec, HM 16.20, providing consistent overall bitrate savings when tested on video sequences at various resolutions and diverse content[1].

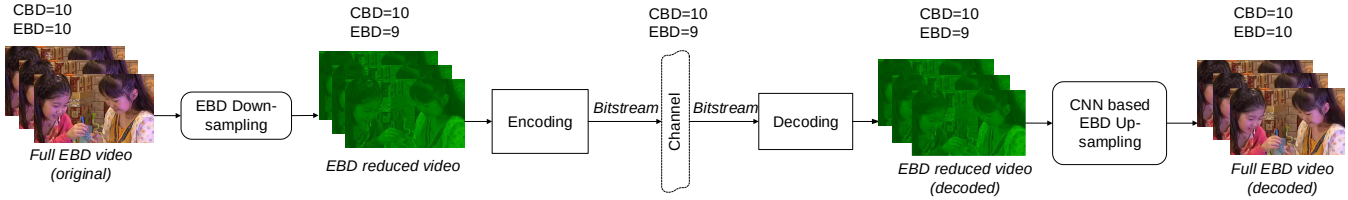


Fig. 1: Diagram of the EBDA-CNN framework for video compression, where The bit depth changes are shown at different stages.

The remainder of this paper is organised as follows. Section II presents the EBDA-CNN coding framework and its key components in detail, while compression results are provided and discussed in Section III. Finally, Section IV concludes the paper and outlines future work.

II. Proposed Algorithm

The framework of the proposed video codec based on the EBDA-CNN approach is shown in Fig.1. In order to clearly describe the proposed method, two bit depth related terms need to be defined in advance.

- **Coding bit depth (CBD)**: this is the definition conventionally used to describe pixel bit depth in video coding and is defined as *InternalBitDepth* in HEVC HM codecs. This remains constant within the coding loop in this work.
- **Effective bit depth (EBD)**: this is the actual bit depth used to represent the video content, which is lower than or equals CBD in the proposed approach.

A. EBD Down-sampling

The full EBD (EBD=CBD) video frames (both luma and chroma channels) are firstly down-sampled (by 1 bit in this paper) through bit shifting.

B. Encoding and Decoding

The video frames with reduced EBD are then encoded by the host encoder using full CBD (CBD=EBD+1). At the decoder, the reduced EBD video frames are then reconstructed from the bitstream. During both encoding and decoding, the host codec does not receive any information on the EBD changes, and processes video content based on its CBD. In order to obtain similar bit rates with full EBD coding (without adaptation) for meaningful comparison (to calculate Bjøntegaard Delta measurements [20] results and estimate complexity), a fixed QP offset of -6 is applied on the initial base QP value when adaptation is enabled.

C. The CNN Architecture Employed

At the decoder, a deep CNN has been used to reconstruct full EBD; its architecture is shown in Fig. 2. The CNN is designed to take 96×96 compressed colour image blocks (RGB colour space) with reduced EBD (EBD=CBD-1) as input, with the target to produce the respective original colour image block with full EBD (EBD=CBD) as output. The developed CNN architecture is similar to that of SRResNet [21] for super-resolution. Its initial convolutional layer is followed by a succession of 16 residual blocks, each of them containing a parametric ReLU as the activation function and two convolutional layers. In each residual block, there is also a skip connection

[1]An early version of this work was contributed by the University of Bristol (JVET-J0031) [18] to the JVET "Call for proposals" for Versatile Video Coding (VVC) [19].

between the input of the first convolutional layer and the output of the second. An additional skip connection is also employed between the output of all 16 residual blocks and the output of the first convolutional layer. In the end, after a single convolutional layer (with a Tanh activation), a final skip connection is employed between the initial input and the output of the last convolution layer to produce the final output image block.

It is noted that all the convolutional layers have small 3×3 kernels and 64 feature maps, and a stride value equal to 1. The exception is the last convolutional layer, which has only 3 feature maps. Comparing to SRResNet, the employed architecture does not contain any batch normalisation (BN) layers. This is based on the recent research on the use of BN, which reported that it could affect the overall performance due to the decrease in variability of image features after normalisation [22]. Additional modifications include the use of l_1 loss instead of the popular l_2 loss, inspired by recent work which found that this could achieve higher quality in image reconstruction tasks [22, 23].

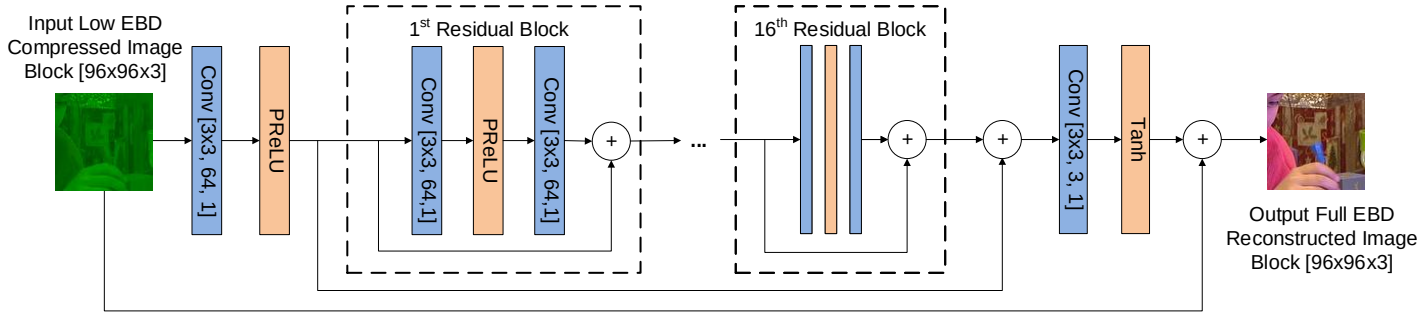


Fig. 2: Proposed CNN architecture for EBD up-sampling.

D. Network Training using Compressed Content

Eighty video sequences, originally from the Harmonic Inc video database [24], were used for training the CNN. These sequences are at different resolutions including 3840×2160, 1920×1080, 960×540 and 480×270. Lower resolutions were obtained from their 2160p sources using Lanczos3 filters. Each sequence has a frame rate of 60 fps, coding bit depth (CBD) of 10 bit, and 64 frames in total[2]. All frames were converted to 9 bit through bit shifting, and compressed by HEVC HM 16.20 under the Joint Video Exploration Team (JVET) Common Test Conditions (CTC) [25] using the Random Access (RA) configuration for four different initial base QP values, 22, 27, 32 and 37 (the fixed QP offset of -6 was applied during encoding). These reconstructed videos with reduced EBD alongside their corresponding full EBD uncompressed sequences were then used as training inputs to the CNN and output targets respectively.

Since this CNN is used for HEVC compressed content, training materials were further sub-grouped as four sets corresponding to four different base QP values, 22, 27, 32 and 37. Based upon them, four different CNN models (model_1, model_2, model_3 and model_4, corresponding to training subgroups QP22, QP27, QP32 and QP37 respectively) were then obtained and are used for different initial (before applying the offset) base QP values ($\mathrm{QP}_{\mathrm{base}}$) in evaluation:

$$\mathrm{CNNs} = \begin{cases} \mathrm{model}_1, \text{ if } & \mathrm{QP}_{\mathrm{base}} \leq 24.5 \\ \mathrm{model}_2, \text{ if } & 24.5 < \mathrm{QP}_{\mathrm{base}} \leq 29.5 \\ \mathrm{model}_3, \text{ if } & 29.5 < \mathrm{QP}_{\mathrm{base}} \leq 34.5 \\ \mathrm{model}_4, \text{ if } & \mathrm{QP}_{\mathrm{base}} \geq 34.5 \end{cases} \tag{1}$$

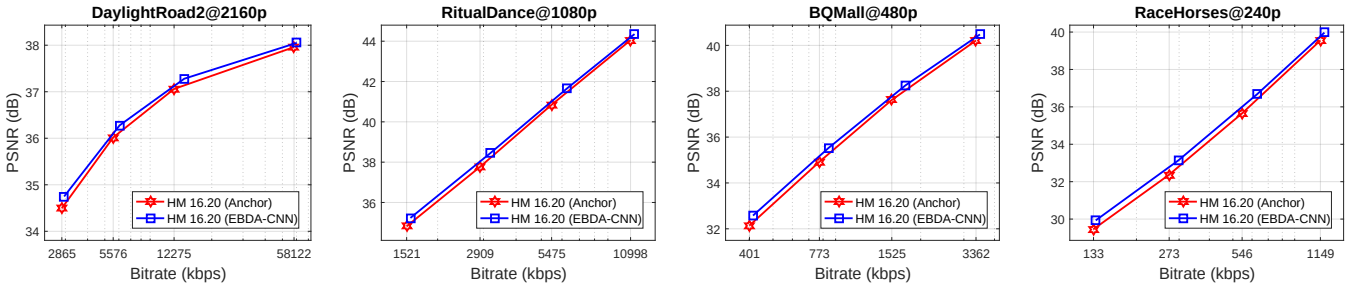
During CNN training, the input/target frames from each subgroup were randomly selected and split into 96×96 pixel blocks. Data augmentation was also applied through block rotation to achieve enhanced model generalisation. This results in a total number of 15,000 pairs of input/target image blocks for training each of four CNN models. The CNN was built and trained using Tensorflow (1.8.0) [26], and the following training parameters were employed: Adam optimisation [27], batch size of 16, learning rate of 10^{-4}, weight decay of 0.1 and 200 epochs.

[2]In this paper, EBDA-CNN is solely targeting 10 bit standard dynamic range video content. The same workflow can be repeated for different original EBD and dynamic range material.

E. CNN-based EBD Up-sampling

In the evaluation phase, when applying CNN-based EBD Up-sampling on large compressed frames, each frame is also split into 96×96 overlapping blocks as the input of the respective CNN model (according to the base QP), with an overlap size of 4 pixels. The full EBD blocks produced by the CNN (output) are then aggregated in the same way to form a final reconstruction frame.

III. Results and Discussion



(a) DaylightRoad2 (b) RitualDance (c) BQMall (d) RaceHorses

Fig. 3: Rate-PSNR curves for four different test sequences: *DaylightRoad2*, *RitualDance*, *BQMall* and *RaceHorses*. Here a logarithmic scale is used at the bitrate axis for better illustration.

The proposed CNN-based effective bit depth adaptation (EBDA-CNN) approach has been integrated with the HEVC reference software (HM 16.20), and was fully tested under JVET CTC [25] using the Random Access configuration (Main10 profile)[3]. The initial base QP values ranged from 22 to 37 with a interval of 5. The parameter *InternalBitDepth* in HEVC HM is fixed as 10 (bit). All the test sequences used are from JVET CTC video class A1, A2, B, C and D, which are different from those used in CNN training.

The compression performance of the EBDA-CNN integrated codec has been compared with the original HEVC HM 16.20. Results for adaptation with simple EBD up-sampling (without using the CNN) were also generated (denoted as EBDA-w/o CNN) in order to provide an additional benchmark for the proposed method. EBDA-w/o CNN reconstructs the decoded EBD reduced video frames to full EBD by simply bit shifting 9 bit to 10 bit. All the results are based on Bjøntegaard Delta (BD) measurements [20] over all frames using PSNR quality metric (only luma channel was calculated).

The complexity figures of the proposed EBDA-CNN based video codec were also estimated based on its average execution time (normalised to original HEVC HM 16.20). Both encoding and decoding were executed on a shared cluster, BlueCrystal Phase 4 [28] based in the University of Bristol, which contains 525 Lenovo nx360 m5 compute nodes. Each node has 14 core 2.4 GHz Intel E5-2680 v4 (Broadwell) CPUs with 128GB of RAM. The decoding jobs were run on GPU nodes with an additional graphic card NVIDIA P100.

A. Compression Performance

Table I shows the compression performance of the proposed EBDA-CNN approach for JVET CTC test sequences when integrated into HEVC HM 16.20, compared with the original HM 16.20 and EBDA-w/o CNN. All the BD results reported in Table I are based on the calculation when original HEVC HM 16.20 is used as the anchor codec for benchmarking. Rate-PSNR curves of the original HM and the proposed EBDA-CNN for four selected sequences, *DaylightRoad2* (Class A2), *RitualDance* (Class B) , *BQMall* (Class C) and *RaceHorses* (Class D) are shown in Fig. 3.

It can be observed that, without using CNN-based up-sampling, EBDA-w/o CNN does not provide any significant improvement in coding efficiency. The average BD-rate is only -0.6%, and it performs worse than the anchor in many cases, with positive BD-rate values up to 2.3% (coding efficiency loss). On the other hand, for EBDA-CNN, although the actual savings still vary according to content type, coding gains have been achieved for all test

[3]The proposed EBDA-CNN has not been trained or tested on the VVC VTM software, as VVC is still under development and its bitstream format has not been finalised.

TABLE I: Bjøntegaard Delta results for tested sequences.

Class-Sequence	EBDA-CNN		EBDA-w/o CNN	
	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR
A1-Campfire	-11.4%	+0.18dB	-9.8%	+0.17dB
A1-FoodMarket4	-4.2%	+0.13dB	+1.4%	-0.04dB
A1-Tango2	-6.1%	+0.09dB	-0.0%	+0.01dB
A2-CatRobot1	-7.9%	+0.14dB	-0.1%	+0.01dB
A2-DaylightRoad2	-8.5%	+0.11dB	+0.6%	+0.00dB
A2-ParkRunning3	-12.0%	+0.50dB	-7.8%	+0.32dB
Class A	**-8.4%**	**+0.19dB**	**-2.6%**	**+0.08dB**
B-BQTerrace	-7.5%	+0.12dB	-0.2%	+0.02dB
B-BasketballDrive	-7.4%	+0.17dB	-1.7%	+0.04dB
B-Cactus	-1.4%	+0.03dB	-0.3%	+0.00dB
B-MarketPlace	-2.7%	+0.07dB	+2.3%	-0.06dB
B-RitualDance	-4.4%	+0.21dB	+1.2%	-0.05dB
Class B	**-4.7%**	**+0.12dB**	**+0.3%**	**-0.01dB**
C-BQMall	-5.6%	+0.21dB	+1.6%	-0.06dB
C-BasketballDrill	-6.0%	+0.26dB	+1.2%	-0.05dB
C-PartyScene	-3.7%	+0.16dB	+1.6%	-0.06dB
C-RaceHorses	-6.3%	+0.24dB	-2.0%	+0.08dB
Class C	**-5.4%**	**+0.22dB**	**+0.6%**	**-0.02dB**
D-BQSquare	-7.0%	+0.25dB	+2.1%	-0.07dB
D-BasketballPass	-7.2%	+0.36dB	-0.8%	+0.05dB
D-BlowingBubbles	-4.5%	+0.18dB	+1.3%	-0.05dB
D-RaceHorses	-6.9%	+0.34dB	-1.2%	+0.06dB
Class D	**-6.4%**	**+0.28dB**	**-0.1%**	**+0.01dB**
Overall	**-6.4%**	**+0.20dB**	**-0.6%**	**+0.02dB**

sequences, with an average BD-rate of -6.4%, and BD-PSNR of 0.2dB. It is also observed from Fig. 3 that the improvement of coding performance is consistent across the whole tested QP range for these four sequences (this is also valid for other test videos). For *Campfire* and *ParkingRunning3* sequences, both EBDA-CNN and EBDA-w/o CNN can offer significant coding gains (from 7.8% to 12.0%). This has been previously reported in JVET meetings [29] but only for chroma QP modification. Further investigation should be conducted to explore why these two sequences are more sensitive.

B. Complexity Analysis

In terms of the complexity of the EBDA-CNN video codec, the average encoding time when it is integrated into HM 16.20 is 1.02 times that of the original HM 16.20, although much lower QP values (with a QP offset of -6) are employed to produce similar bit rate to the anchor for each rate point. However, due to the use of the CNN for EBD up-sampling, the average decoding time is 69.3 times that of HM.

IV. Conclusions

In this paper, an effective bit depth adaptation (EBDA-CNN) approach has been proposed for video coding. It reduces effective bit depth (EBD) by 1 bit before encoding, and reconstructs full bit depth at the decoder using a deep Convolutional Neural Network (CNN) based up-sampling method. This approach has been integrated into HEVC reference codec HM 16.20 and fully evaluated on JVET CTC test sequences. The results show that consistent coding gains can be achieved for all tested sequences, with an average BD-rate of -6.4%. Future work will focus on reducing the complexity of the CNN for EBD up-sampling and its application on higher dynamic range (bit depth) content.

V. References

[1] CISCO, "CISCO visual networking index: forecast and methodology, 2017–2022," November 2018.

[2] B. Bross, J. Chen, and S. Liu, "Versatile video coding (draft 4)," in *the JVET meeting*, no. JVET-M1001. ITU-T and ISO/IEC, 2019.

[3] J. Chen, Y. Ye, and S. Kim, "Algorithm description for Versatile Video Coding and test model 4 (VTM 4)," in *the JVET meeting*, no. JVET-M1002. ITU-T and ISO/IEC, 2019.

[4] ITU-T Rec. H.265, *High efficiency video coding*, ITU-T Std., 2015.

[5] ITU-T Rec. H.264, *Advanced Video Coding for Generic Audiovisual Services*, ITU-T Std., 2005.

[6] Y. Li, D. Liu, H. Li, L. Li, F. Wu, H. Zhang, and H. Yang, "Convolutional neural network-based block up-sampling for intra frame coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 9, pp. 2316–2330, 2018.

[7] J. Dong and Y. Ye, "Adaptive downsampling for high-definition video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 3, pp. 480–488, 2014.

[8] Q. Huang, S. Y. Jeong, S. Yang, D. Zhang, S. Hu, H. Y. Kim, J. S. Choi, and C.-C. J. Kuo, "Perceptual quality driven frame-rate selection (PQD-FRS) for high-frame-rate video," *IEEE Trans. on Broadcasting*, vol. 62, no. 3, pp. 640–653, 2016.

[9] Z. Ma, M. Xu, Y.-F. Ou, and Y. Wang, "Modeling of rate and perceptual quality of compressed video as functions of frame rate and quantization stepsize and its applications," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 22, no. 5, pp. 671–682, 2012.

[10] T. Laude and J. Ostermann, "Deep learning-based intra prediction mode decision for HEVC," in *Picture Coding Symposium (PCS), 2016*. IEEE, 2016, pp. 1–5.

[11] C.-H. Yeh, Z.-T. Zhang, M.-J. Chen, and C.-Y. Lin, "HEVC intra frame coding based on convolutional neural network," *IEEE Access*, vol. 6, pp. 50 087–50 095, 2018.

[12] J. Liu, S. Xia, W. Yang, M. Li, and D. Liu, "One-for-all: Grouped variation network based fractional interpolation in video coding," *IEEE Transactions on Image Processing*, 2018.

[13] H. Zhang, L. Song, Z. Luo, and X. Yang, "Learning a convolutional neural network for fractional interpolation in HEVC inter coding," in *Visual Communications and Image Processing (VCIP), 2017 IEEE*. IEEE, 2017, pp. 1–4.

[14] S. Kuanar, C. Conly, and K. R. Rao, "Deep learning based hevc in-loop filtering for decoder quality enhancement," in *2018 Picture Coding Symposium (PCS)*. IEEE, 2018, pp. 164–168.

[15] W.-S. Park and M. Kim, "Cnn-based in-loop filtering for coding efficiency improvement," in *Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), 2016 IEEE 12th*. IEEE, 2016, pp. 1–5.

[16] M. Afonso, F. Zhang, and D. R. Bull, "Video compression based on spatio-temporal resolution adaptation," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 29, no. 1, pp. 275–280, January 2019.

[17] ——, "Spatial resolution adaptation framework for video compression," in *Proc. SPIE, Applications of Digital Image Processing XLI*, vol. 10752. International Society for Optics and Photonics, 2018, p. 107520L.

[18] D. Bull, F. Zhang, and M. Afonso, "Description of SDR video coding technology proposal by University of Bristol (JVET-J0031)," in *the JVET meeting*, no. JVET-J0031. San Diego, US: ITU-T and ISO/IEC, April 2018.

[19] A. Segall, V. Baroncini, J. Boyce, J. Chen, and T. Suzuki, "Joint call for proposals on video compression with capability beyond hevc," in *the JVET meeting*, no. JVET-H1002. Macao, CN: ITU-T and ISO/IEC, October 2017.

[20] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," in *13th VCEG Meeting*, no. VCEG-M33. Austin, Texas, USA: ITU-T, April 2001.

[21] C. Ledig and *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 105–114.

[22] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *The IEEE conference on computer vision and pattern recognition (CVPR) workshops*, vol. 1, no. 2, 2017, p. 4.

[23] J. Johnson, A. Alahi, and F.-F. Li, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*. Springer, 2016, pp. 694–711.

[24] Harmonic, “Harmonic free 4K demo footage.” [Online]. Available: https://www.harmonicinc.com/free-4k-demo-footage/#4k-clip-center

[25] F. Bossen, J. Boyce, X. Li, V. Seregin, and K. Sühring, “Jvet common test conditions and software reference configurations for sdr video,” in *the JVET meeting*, no. JVET-M1001. ITU-T and ISO/IEC, 2019.

[26] A. Martín *et al.*, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[27] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” in *International Conference on Learning Representations*, 2015.

[28] University of Bristol, “BlueCystal Phase 4.” [Online]. Available: https://www.acrc.bris.ac.uk/acrc/phase4.htm

[29] G. Sullivan and J.-R. Ohm, “Meeting report of the 10th meeting of the Joint Video Experts Team (JVET),” in *the JVET meeting*, no. JVET-J_Notes_d. ITU-T, ISO/IEC, April 2018.