TOWARDS AN AI-ENHANCED VIDEO CODING STANDARD

L. Chiariglione (MPAI, IT); R. Iacoviello (Rai, IT); A. Fiandrotti (Università di Torino, IT); A. Artusi , M. Angelini (CYENS, CY), A. Mosca, G. Ballocca (Sisvel Technology, IT), F. Gissi, A. Martinelli (Kebula, IT)

ABSTRACT

This paper describes the ongoing activities of the Enhanced Video coding (EVC) project of the Moving Picture, Audio and Data Coding by Artificial Intelligence (MPAI). The project investigates how the performances of existing codecs can be improved by enhancing or replacing specific encoding tools with AI-based counterparts. The MPEG EVC codec baseline profile has been chosen as reference as it relies on encoding tools that are at least 20 years mature yet has compression efficiency close to HEVC. A framework has been developed to interface the encoder/decoder with neural networks, independently from the specific learning toolkit, simplifying experimentation. So far, the EVC project has investigated the intra prediction and the super resolution coding tools. The standard intra prediction modes have been integrated by a learnable predictor: experiments in standard test conditions show rate reductions for intra coded frames in excess of 4% over the reference. The use of super resolution, a state-of-the-art deep-learning approach named Densely Residual Laplacian Network (DRLN), at the decoder side have been found to provide further gains, over the reference, in the order of 3% in the SD to HD context.

INTRODUCTION

MPAI is an international, unaffiliated, non-profit standards developing organisation that has the mission to develop Artificial Intelligence (AI) enabled data coding standards. Its standard development process corrects other standardisation bodies' shortcomings, by adding a clear Intellectual Property Rights (IPR) licensing framework. MPAI has already developed the AI Framework (AIF) standard (MPAI-AIF), specifying AIF as an environment capable of managing the life cycle of AI Workflows (AIW) and their components called AI modules (AIMs). AIWs are defined by their function, i.e. an MPAI-specified Use Case, the syntax and semantics of the input and output data and the AIM topology. Similarly, AIMs are defined by their function (e.g. motion compensation) and the syntax and semantics of the input and output data, but not the AIM internals. By basing its standards on AIMs, implementers of MPAI standards can have a low entry barrier to an open competitive market for their implementations because application implementers can find the AIMs they need on the open market. The MPAI-AIF standard is currently being extended by adding the capability to access trusted services.

Since the day MPAI was announced, there has been considerable interest in the application of AI to video. Video contents nowadays accounts for more than 70% of

Internet traffic volume [Cisco, (1)], hence the interest into efficient video coding technologies able to cope with tomorrow bandwidth-demanding video services (4K video, immersive contents, etc.).

Existing video coding standards used in Internet streaming or aerial broadcasting over the air or cable rely on a clever combination of hand-designed encoding tools, each bringing its own contribution to the overall codec performance as shown in Figure 1.

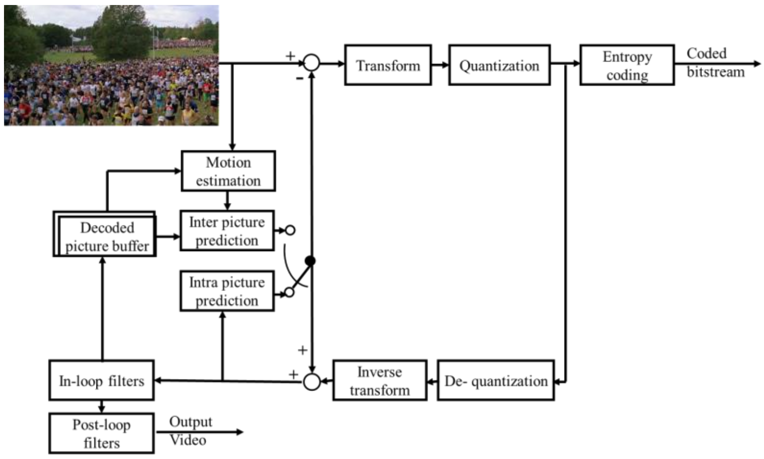

Figure 1: Hybrid video codec schema

This can be achieved by predicting the picture from neighbouring data within the same picture (known as intra-prediction) or from data previously signalled in other pictures (known as inter-prediction). Intra-prediction uses previously decoded sample values of neighbouring samples to assist in the prediction of current samples.

The residual signal is then transformed via discrete cosine transform, allowing low-pass filtering in the transformed domain. Coefficient decimation and the subsequent quantisation is the lossy part of the compression process that allows to reduce the high frequency rate while keeping the resulting artefacts bearable to the human observer.

The resulting signal is entropy encoded, which is a lossless form of compression.

Within the encoder, when some sort of prediction is enabled, the encoded signal may be reconstructed through a de-quantisation and inverse transformation step and the input visual data is reconstructed by adding the predicted signal. Filters, such as a deblocking filter and a sample adaptive offset filter are used to improve the visual quality. The reconstructed picture is stored for future reference in a reference picture buffer to allow exploiting the similarities between two pictures.

The motion estimation process evaluates one or more candidate blocks by minimizing the distortion compared to the current block. The residual between the current and optimal block is used by the motion compensation, which creates a prediction for the current block. The inter-prediction exploits redundancies between pictures of visual data. Reference pictures are used to reconstruct pictures that are to be displayed, resulting in a reduction in the amount of data required to be transmitted or stored.

However, since resolution and frame rates are increasing at the same time, relying on hardware advances is no longer sufficient for some applications. Over the past years, the research community has investigated the recent developments in Artificial Intelligence (AI) and Machine Learning (ML), to push the boundaries and deliver industry-leading video quality and hardware efficiency.

There are two main approaches in the AI-based video coding research community: 1) one approach introduces learning based algorithms combined with traditional image video codec, trying to replace one coding block with the AI-based one; 2) an End-to-End (E2E) approach which is mainly focused on replacing the entire chain with a pure deep learning based compression.

Both research directions are being explored within MPAI by the End-to-End Video Coding group, (EEV), and the Enhanced Video Coding group, (EVC), respectively. This document details the recent activities of the EVC group.

The primary goal of MPAI-EVC is to enhance the performance of traditional video codecs by integrating AI-based coding tools. The first step is the MPAI-EVC Evidence Project [Chiariglione et al. (2)] with the intent to demonstrate that AI tools can improve the MPEG-5 EVC efficiency by at least 25%. Two main tools have been investigated, namely the intra prediction enhancement and the super resolution. The EVC reference schema is depicted in Figure 2.

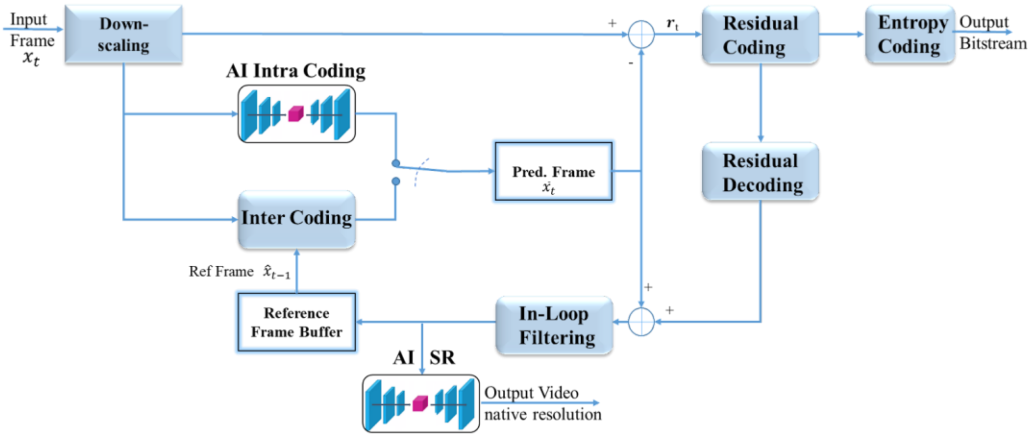

Figure 2: Reference schema for EVC Evidence project

A parallel activity to the MPAI-EVC Evidence Project is the MPAI End-to-End Video Coding project (MPAI-EEV) aiming to address the needs of the many who need not only environments where academic knowledge is promoted but also a body that develops common understanding, models and eventually standards-oriented End-to-End video coding solutions. MPAI-EEV can cover the medium-to-long term video coding needs. Currently the group has developed a study of the state of the art of end-to-end video coding and has decided to start from the OpenDVC [Yang et al. (3)] software to develop a reference model that will be used for collaborative investigations.

The rest of the paper describes in detail the activities of the EVC project with the Intra prediction and Super resolution tools.

INTRA PREDICTION TOOL

The first tool investigated by the EVC project is intra prediction tool, with the goal of integrating a learnable intra predictor within the EVC encoder. Intra-frame prediction leverages the spatial correlation within the same picture generating a predictor for the Coding Unit (CU) to be encoded by extrapolating pixel values from a previously encoded neighbourhood. The predicted block is then subtracted from the original block, producing a residual block that is transformed, quantized and entropy coded before being inserted into the bitstream with the predictor mode index. At the decoder side, the signalled predictor is generated from the decoded context and then the residual is decoded, added to the predictor, recovering the encoded block. The rationale behind intra prediction is that encoding the residual requires fewer bits than encoding the original block. The better the predictor, i.e. the closer to the block to be encoded, the lower the residual rate and the higher the coding efficiency. The MPEG-5 EVC base profile offers 5 intra prediction modes: DC, horizontal, vertical and two diagonal modes and for each CU, the encoder selects the intra mode that minimises the residual rate, which may be then put into competition with other modes. We addressed the problem of predicting a block from its context as an image inpainting problem, i.e. recovering pixels of an image that are unavailable due to, e.g. occlusions. Recently, deep neural networks [LeCun et al. (4)] have shown to outperform classic inpainting methods thanks to their ability to learn complex nonlinear functions. We leverage recent advances in deep generative models recasting the task of generating an intra predictor as a hole inpainting problem [Pathak et al. (5)] following promising works with different codecs [Wang et al. (6), Dumas et al. (7)]

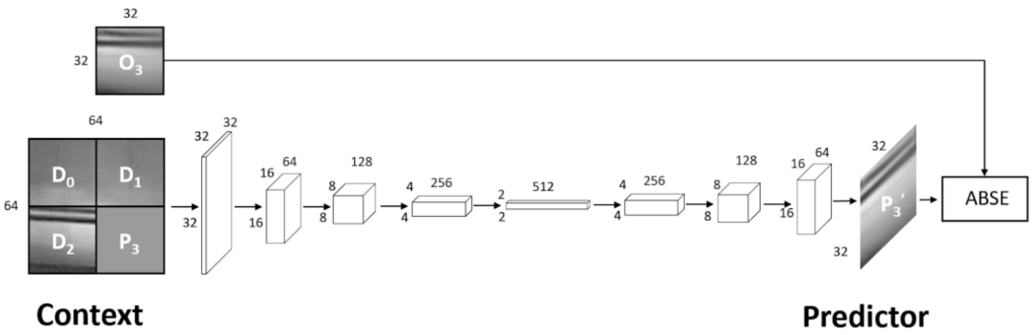

Figure 3: Architecture and procedure for training the convolutional autoencoder used to generate a learnable intra predictor. In this example, a 32x32 predictor is generated from a 64x64 context.

Figure 3 shows the convolutional autoencoder architecture that we adapted towards the task of generating an intra predictor. For the sake of simplicity, we exemplify the case of generating a 32x32 predictor from a 64x64 context, however similar considerations hold for the other CU sizes supported by EVC (16x16, 8x8, 4x4 CUs). The autoencoder receives in input a 64x64 patch representing the encoded context also available at the decoder (D0, D1, D2). The 32x32 bottom-right corner (P3) is the predictor to be generated, i.e. the area of image to inpaint. The autoencoder is trained to output a 32x32 patch (P3’) that represents our learnable predictor and should be a reasonable approximation of the original block O3. The autoencoder includes 5 pooling layers and relies on LReLUs as activations of the hidden layers. The autoencoder input is normalised in the [-1, 1] range and the output layer has a TanH activation.

The autoencoder is trained by minimising the absolute error between P3’ and O3 on a dataset of 800 images of different resolution and content type randomly sampled from the AROD dataset [Schwarz et al. (8)]. While these images are JPEG compressed, they are very high quality, which is almost equivalent to training over uncompressed images for the purpose of training a neural network. Contrary to similar approach, we train one single network instance regardless of the encoding QP, hence simplifying the complexity of the proposed approach.

From each image, a 64x64 patch is cropped at a random position, the patch is then randomly flipped horizontally and vertically, followed by a 90 degrees random rotation. Our experiments showed that this form of augmentation is key to prevent the autoencoder from overfitting on the training data. The bottom right 32x32 corner of the patch represents the O3 original CU to recover, whereas the rest of the patch represents the (D0, D1, D2) context. Prior to training, the P3 corner is filled with black pixels to represent the image area to be inpainted. The autoencoder is trained with SGD with a learning rate of 0.01 and over batches of 64 patches.

Once the autoencoder has been trained, it is interfaced with the EVC encoder as follows. First, a networked server process is started, loads the trained autoencoder into the GPU memory, sets up an UDP socket in listening mode and awaits for incoming messages. The EVC encoder was modified so that the mode 0 intra predictor (DC mode) is repurposed to handle the predictor generated by the autoencoder. For each intra-coded CU, the EVC encoder was modified to send to the server the 64x64 decoded context (D0, D1, D2, P3). The server inputs such context to the trained autoencoder and returns the 32x32 output P3', i.e. the learned predictor, to the encoder via the same UDP socket. The UDP socket scheme allows one to easily experiment with different neural network frameworks (PyTorch, TensorFlow, Keras, etc.) without modifying the encoder, thus simplifying the experiments. Finally, the modified EVC encoder replaces the DC predictor with the autoencoder generated predictor and the encoding proceeds as usual, i.e. by putting the learned predictor in competition with the other 4 EVC intra predictors.

We point out that no modifications are required to the signalling since the DC mode is simply replaced with our predictor, and the bitstream remains fully decodable under the reasonable assumption that the EVC decoder has available the same autoencoder used by the EVC encoder.

We experimented encoding the first frame of the well known JVET CTC sequences in Table 1 in the 22-42 QP range and the results are shown in Table1 (32x32, 16x16 and 8x8 CUs only benefit from the learned predictor, while 4x4 CUs rely on the original DC predictor).

		Proposed		**Oracle**	
Sequence	**Class**	**BDRate [%]**	**BDPSNR [dB]**	**BDRate [%]**	**BDPSNR [dB]**
Campfire	Class A 3840x2160 60/50 fps 10 bpp	-1.51	0.06	-3.33	0.12
CatRobot		-5.54	0.18	-7.02	0.23
DaylightRoad2		-6.50	0.16	-7.79	0.20
FoodMarket4		-8.62	0.26	-9.85	0.29
ParkRunning3		-1.63	0.10	-2.67	0.17
Tango2		-7.30	0.14	-8.86	0.17
Average		**-5.18**	**0.15**	**-6.59**	**0.20**
BQTerrace	Class B	-3.11	0.19	-4,30	0,27

BasketballDrive	1920x1080 60/50 fps 10/8 bpp	-7.66	0.23	-8,87	0,27
Cactus		-4.96	0.21	-6,41	0,27
MarketPlace		-4.71	0.17	-7.00	0,26
RitualDance		-8.51	0.47	-10,66	0,59
Average		**-5.79**	**0.25**	**-7,45**	**0,33**
BQMall	Class C 832x480 60/50/30 fps 8 bpp	-2.68	0.17	-3,6	0,22
BasketballDrill		-4.32	0.22	-5,2	0,27
PartyScene		-1.32	0.09	-1,91	0,14
RaceHorsesC		-3.70	0.24	-4,78	0,31
Average		**-3.00**	**0.18**	**-3,87**	**0,23**
BQSquare	Class D 416x240 60/50/30 fps 8 bpp	-0.51	0.04	-0,9	0,08
BasketballPass		-2.03	0.12	-2,66	0,16
BlowingBubbles		-2.14	0.13	-2,85	0,17
RaceHorsesD		-2.28	0.16	-3,09	0,22
Average		**-1.74**	**0.11**	**-2,38**	**0,16**
FourPeople	Class E 1280x720 60 fps 8 bpp	-8.26	0.51	-9,82	0,61
Johnny		-7.53	0.35	-9,25	0,43
KristenAndSara		-6.32	0.37	-7,71	0,45
Average		**-7.37**	**0.41**	**-8,93**	**0,5**
ArenaOfValor	Class F Screen content Multiresolution 60 fps 8 bpp	-2.57	0.16	-3,97	0,24
BasketDrillText		-3.14	0.18	-4,6	0,25
SlideEditing		-0.20	0.03	-0,72	0,1
SlideShow		-0.33	0.04	-0,82	0,1
Average		**-1.56**	**0.10**	**-2,53**	**0,17**
Grand Average	All of the above	**-4,13**	**0,19**	**-5,33**	**0,25**

Table 1: Results obtained by replacing the DC mode with a convolutional autoencoder generated intra predictor, QP 22-42 range.

The experiments report BD-Rate reductions in excess of 8% and BD-PSNR improvements close to 0.5 dB for some sequences. The experiments show gains especially for sequences above 720p: we attribute that to the fact that most of the training images are above 600 pixels in height. We hypothesise that the addition of smaller images to the training set would boost the performance on classes C and D. Lowest performance is achieved for screen contents (Class F), a non-unexpected result if we consider that our training set contains no computer screen images. A visual inspection of the decoded sequences shows no perceivable artefacts despite the learned intra predictor.

Finally, the two rightmost columns of the table report the performance of the scheme when a sixth hypothetical Oracle predictor is put into competition with the standard 5 intra EVC modes, rather than replacing the DC predictor. While the extra signalling is not present and so the bitstream is not decodable, this scheme shows extra gains above 1%, suggesting that proper implementation of our learned intra predictor may yield further gains [Helle et al. (9)] if put into competition with the other modes.

Concerning the complexity of the proposed intra tool, our autoencoder includes about 3M learnable parameters. While for the sake of simplicity the autoencoder retains the same architecture regardless the CU size, for CUs smaller than 32x32 some convolutional layers can be dropped, reducing such complexity. About the end-to-end inference time of our prototype implementation, it is about 22ms per 32x32 CUs on a NVIDIA T4 GPU. We point out that such number includes the overhead associated with the UDP socket communications that would be absent in an optimized implementation relying on direct GPU-coupling or FPGA acceleration.

SUPER RESOLUTION TOOL

The super resolution tool is used as an upsampling step in the EVC decoding system and implemented as a post-loop filter. For better and easier control of the possible improvement of the overall performances of the EVC decoding system, an existing super resolution deep-learning based approach has been applied to the decoded picture allowing to recover the full resolution frame.

Among several state-of-the-art super resolution approaches, we selected the well known Densely Residual Laplacian Network (DRLN) [Anwar et al. (10)], which has been proven to provide best performances among the existing approaches. This architecture is employed as an up-sampler whenever the input sequence has been downsampled, into the decoding system. The overall deep-learning structure of the DRLN approach is depicted in Figure 4:

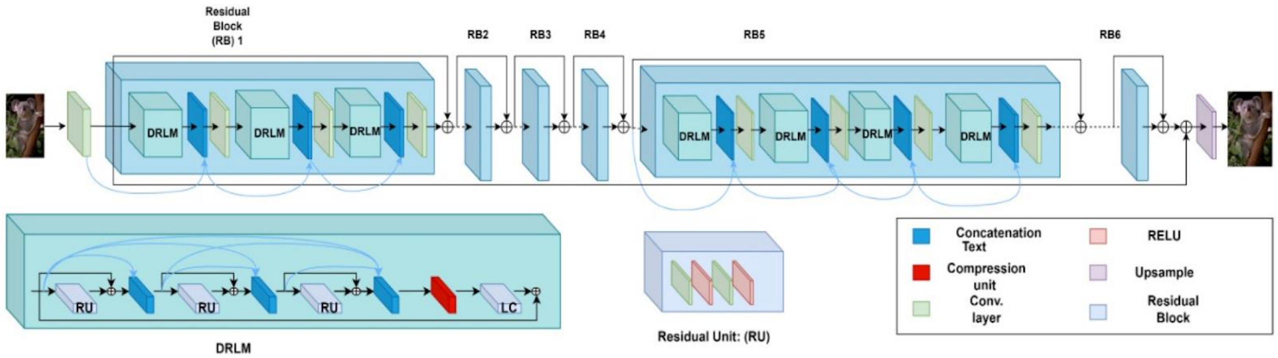

Figure 4: The detailed network architecture of the DRLN model. The top figure shows the whole network architecture consisting of six cascaded residual blocks (RB). The bottom figure shows the internal structure of sub-components i.e. densely residual laplacian module (DRLM) and Residual Units(RU). Courtesy of Asfa Jamil, after [Helle et al. (9)]

Our experiments have been concentrated in demonstrating the capabilities of the DRLN approach to improve the performances of the EVC coding system, for the upsampling from SD to HD resolution (upsampling of a factor of 2). To achieve it, we have first prepared a dataset where the initial 2000 4K images from the Kaggle dataset have been resized to HD (1920x1080) and SD (960x540) resolutions using Lanczos filtering. Images have been converted to the YUV format and EVC encoded, using baseline profile, random access configuration with 32 pictures hierarchical GOP at 15, 22, 30, 45 fixed QPs. The HD counterpart of the frames pair is an encoded EVC bitstream at full HD. This dataset represents the image pool from which the training and validation datasets were extracted.

Using the DRLN directly on the full resolution of the input-output frames, complexity issues could be met, i.e. high computational costs, as well as memory constraints. To avoid these issues, we have designed a cropping strategy for developing training and validation datasets.

The frames have been subdivided in crops of a predefined size, which could fit the memory RAM available in the GPU used in the training step. By doing so, however, a large number of crops were generated and many of them were not carrying significant information making them redundant, and consequently they were increasing the requested training time. To solve this issue, we have employed two strategies, both based on the entropy information of the input frame. This is calculated by estimating at each pixel position (i,j) the entropy of the pixel-values within a 2-dim region centred at (i,j). The first strategy uses a random crop if, and only if, its average entropy exceeds a given threshold. The second strategy selects n crops, of the same size, from the total crops available in each frame. This is based on the importance sampling technique applied to the entropy values distribution of all crops in each frame. A particular attention needs to be given to the right combination of the crop and batch sizes as a trade-off with respect to GPU memory consumptions can be achieved. The table below shows the tested combination:

input - Batch size	GPU memory usage (GiB)
48 - 16	9.7
72 - 16	18.0
96 - 10	19.7
128 - 6	20.0

Table 2: Tested combinations of crop and batch sizes

As described above, two cropping strategies have been employed to efficiently perform the training tasks. The hyperparameters and parameters used during the training phase were the followings: learning rate (lr) 10e-5, batch size 6 and 2 for the crop strategy based on importance sampling, epochs 50, the resolution of the crop input was 128x128, while for the crop output was 256x256, the dataset used was the one with deblocking option activated. The Mean Square Error (MSE) metric was used as a loss function.

The importance sampling performances showed an improvement in terms of PSNR when compared with the random crop strategy. Moreover, better generalisation results are due to the fact that the training and validation sets performance are not showing large discrepancies as in the case of the random crop approach. This was noticed on all the QPs used in the experiment.

Based on these results we have decided to use the importance sampling approach. The strategy adopted to prepare the training and the validation datasets has been to use 80% of the original crop dataset, selected with the importance sampling strategy, as training dataset and the remaining 20% as validation datasets.

We have trained the DRLN on all the QPs for 50 epochs using the same hyperparameters and parameters used for the selection of the cropping strategy.

The training performances for all the QPs are shown in Figure 5. These performances on the training and validation datasets do not have large disagreements in terms of PSNR, suggesting that the training reached good generalisation performances.

Based on these results we have chosen the DRLN weights at epoch 47 for QP 15, epoch 45 for QP 30 and epoch 47 for QP 45 and performed a test on a new set of 8 sequences for understanding its generalisation capabilities, as well as to quantify the gain or the loss in terms of BD-rate performances with respect to the baseline EVC codec.

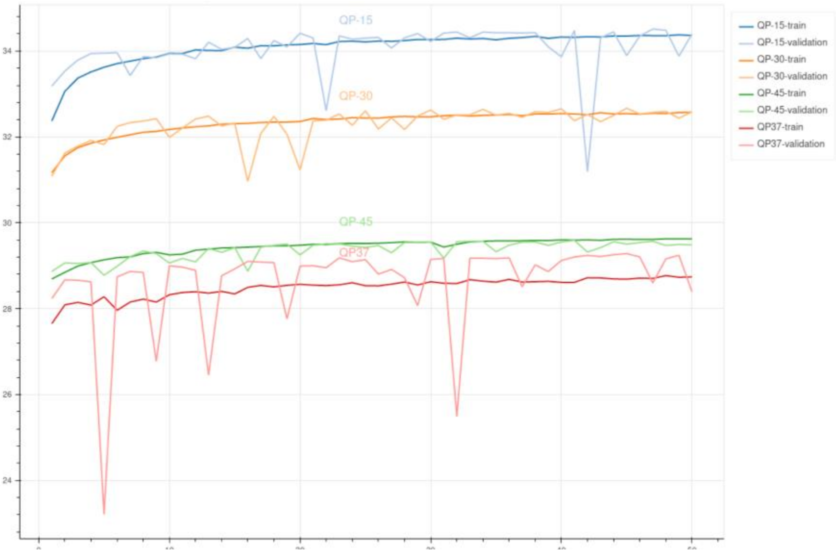

Figure 5: training and validation performances of the SD2HD upsampling on all the QPs.

In order to provide content diversity, three 4K sequences from the SVT archive (Crowd Run, Ducks Take Off and Park Joy) have been resized to HD (1920x1080) and SD (960x540) resolutions using Lanczos filtering. Also 5 HD sequences (one public domain: Rush Hour, and four proprietary sequences: Diego and the Owl, Rome 1, Rome 2 and Talk Show) have been resized to SD (960x540) resolution using Lanczos filtering. The test sequences have been coded using the same encoder configurations as the training set. The BD-rate results of the test reported in Table 3 and Figure 6 show and average improvement in terms of BD-rate of -3.14% for all the test sequences (negative values indicating coding efficiency gain). However, we may notice that there is large variation in terms of performances, across the different streams, e.g. the streams “Rome 2” and “Talk Show” provide a large improvement, while the stream “Diego and Owl” shows a consistent degrading performances when compared to the baseline EVC. This may be related to the generalization capability of DRLN, which it will need to be further investigated.

Sequence	Class	BD-Rate
Crowd Run	Class B 1920x1080 60/50 fps, 8 bpp	-1.24%
Ducks Take Off	Class B 1920x1080 60/50 fps, 8 bpp	2.12%
Park Joy	Class B 1920x1080 60/50 fps, 8 bpp	1.40%
Diego and Owl	Class B 1920x1080 60/50 fps, 8 bpp	8.11%
Rome 1	Class B 1920x1080 60/50 fps, 8 bpp	0.19%
Rome 2	Class B 1920x1080 60/50 fps, 8 bpp	-18.81%
Rush Hour	Class B 1920x1080 60/50 fps, 8 bpp	4.90%
Talk Show	Class B 1920x1080 60/50 fps, 8 bpp	-21.75%
Average: -3.14%		

Table 3: BD-rate performances on all the 8 test sequences

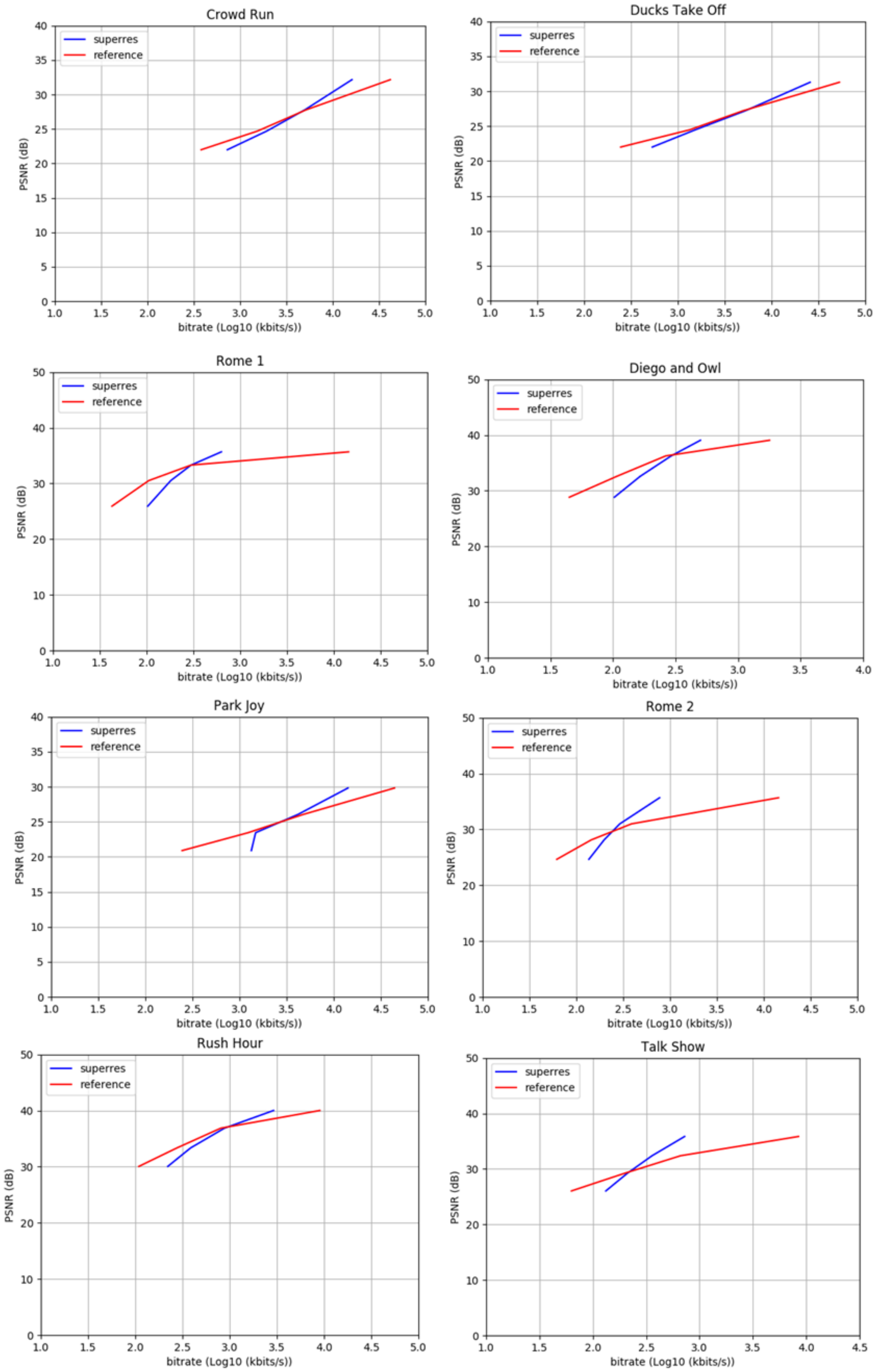

Figure 6: BD-rate performances on all the 8 test sequences

NEXT STEPS AND CONCLUSION

MPAI is a unique environment where experts working on different application areas and data type apply Artificial Intelligence methods to achieve optimal data coding formats.

The current two video coding projects – AI-Enhanced Video Coding (EVC) and End-to-End Video Coding (EEV) are benefitting from the environment.

Concerning the intra predictor tool, the current results of our experiments with a learned predictor constrained to a few CU sizes showed average rate savings in excess of 4% and up to 8% for some sequences. Such gains are expected to improve when our scheme is applied to 4x4 CUs as well. Also, a proper implementation of our learned predictor where it is put into competition with the 5 EVC modes is expected to enable further gains. Moreover, a set of training images better representative of low resolution images and screen contents may improve the performance of our scheme towards this type of content.

The SR tool has shown good overall performances in terms of BD-rate over the standard baseline EVC decoding for the SD2HD task. The model for the task HD24K is currently under training and its preliminary results are also encouraging.
MPAI-EVC is an exploration seeking to demonstrate that AI coding tools can be successfully applied to a traditional video coding architecture and further extend the capability to reduce the bitrate required to represent moving picture information. Once the goal will be achieved – which may be rather soon – MPAI intends to issue a Call for Technologies and develop a standard that satisfies both functional (MPAI, [11]) and commercial requirements. The latter will be embodied in a "Framework Licence". Unlike Fair, Reasonable and Non Discriminatory (FRAND) declarations, the Framework Licence includes terms and conditions without values (dollars, percentages, rates, dates, etc.) and a declaration that the licence will be issued before commercial implementations are available on the market at a total cost in line with the total cost of the licenses for similar data coding technologies considering the market value of the specific standardised technology (MPAI, [12]).

ACKNOWLEDGEMENTS

This work has been supported by funding from the European Union Horizon 2020 research and innovation programme under grant agreement No 739578 for Dr. Artusi and Mr. Angelini. The funding for the work of Dr. Artusi and Mr. Angelini are also complemented by the Government of the Republic of Cyprus through the Directorate General for European Programmes, Coordination and Development.

The work of R. Iacoviello has been partially funded by the European Union's Horizon 2020 research and innovation programme, AI4MEDIA project, under grant agreement No. 951911

REFERENCES

1. Cisco Annual Internet Report (2018–2023) White Paper https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.htm

2. L. Chiariglione et al. “AI-based media coding and beyond”, *Proc. of International Broadcasting Convention IBC’21*, 2021.

3. Yang, R., Van Gool, L., & Timofte, R. (2020). OpenDVC: An Open Source Implementation of the DVC Video Compression Method. http://arxiv.org/abs/2006.15862

4. Y. LeCun, Y. Bengio, G. Hinton. Deep learning. Nature, 521 (7553), (pp 436-444), 2015.

5. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., & Efros, A. A.. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2536-2544), 2016 https://arxiv.org/abs/1611.05203

6. Wang, L., Fiandrotti, A., Purica, A., Valenzise, G., & Cagnazzo, M. (2019, May). Enhancing HEVC spatial prediction by context-based learning. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4035-4039). IEEE.

7. Dumas, T., Roumy, A., & Guillemot, C. (2019). Context-adaptive neural network-based prediction for image compression. IEEE Transactions on Image Processing, 29, 679-693.

8. Schwarz, K., Wieschollek, P., & Lensch, H. P. (2018). Will people like your image? learning the aesthetic space. IEEE winter conference on applications of computer vision (WACV) (pp. 2048-2057), 2018

9. P. Helle, J. Pfaff, M. Schäfer, R. Rischke, H. Schwarz, D. Marpe, T. Wiegand. Intra picture prediction for video coding with neural networks. In 2019 Data Compression Conference (DCC) (pp. 448-457), 2019.

10. S. Anwar, N.Barnes, "Densely Residual Laplacian Super-resolution", IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2020

11. MPAI-EVC Use Cases and Functional Requirements; https://mpai.community/standards/mpai-evc/use-cases-and-functional-requirements/

12. The MPAI Patent Policy; https://mpai.community/about/the-mpai-patent-policy/