A Learnable EVC Intra Predictor Using Masked Convolutions

Gabriele Spadaro, Roberto Iacoviello, Alessandra Mosca, Giuseppe Valenzise, Attilio Fiandrotti

A Learnable EVC Intra Predictor using Masked Convolutions

Gabriele Spadaro[1 [0009−0008−4786−1074]], Roberto Iacoviello[2 [0000−0002−5336−1462]], Alessandra Mosca[3 [2222−−3333−4444−5555]], Giuseppe Valenzise[4 [2222−−3333−4444−5555]], and Attilio Fiandrotti[1 [2222−−3333−4444−5555]]

[1] University of Turin, Italy
{gabriele.spadaro,attilio.fiandrotti}@unito.it
[2] Rai Radiotelevisione italiana
roberto.iacoviello@rai.it
[3] Sisvel Technology S.r.l.
alessandra.mosca@sisveltech.com
[4] CNRS, France
giuseppe.valenzise@l2s.centralesupelec.fr

Abstract. The Enhanced Video Coding (EVC) workgroup of the Moving Picture, Audio and Data Coding by Artificial Intelligence (MPAI) organization aims at enhancing traditional video codecs by improving or replacing traditional encoding tools with AI-based counterparts. In this work, we explore enhancing MPEG Essential Video Coding (EVC) intra prediction with a learnable predictor: we recast the problem as a hole inpainting task that we tackle via masked convolutions. Our experiments in standard test conditions show BD-rate reductions in excess of 6% over the EVC baseline profile reference with some sequences in excess of 12%.

Keywords: EVC · intra prediction · learnable video coding.

1 Introduction

Video content accounts for over 70% of Internet traffic volume [4], hence the interest in efficient video coding technologies. Recently, the trend has been leveraging recent advances in artificial intelligence and deep learning to improve the efficiency of video codecs and two distinct approaches have emerged. The first approach aims at integrating or replacing selected encoding tools of traditional codecs with learnable equivalents. The second approach aims at designing from scratch novel codecs with an end-to-end totally deep learning based architecture. The EVC project of the MPAI community [5] falls in the former category and aims at improving the efficiency of existing video codecs by at least 25% of BD-Rate. The MPEG-5 Essential Video Coding (EVC) [3, 16] baseline profile has been chosen as reference as it relies on encoding tools that are at least 20 years mature, yet it shows compression efficiency comparable to H.265/HEVC [18].

[5] https://mpai.community/standards/mpai-evc/about-mpai-evc/

The MPAI EVC project is currently studying a number of encoding tools based on deep learning, and this paper describes the ongoing activities on the intra prediction tool. Modern video codecs exploit the spatial correlation in pictures predicting each block to be encoded from a previously encoded area (*predictor*) of the same picture. The rationale behind intra prediction is that encoding the difference (the *residual*) between the block pixels and the one associated with its predictor is more efficient than encoding the block pixels themselves. Namely, the closer the predictor pixels are to the block ones, the fewer the bits to encode the residual and so the encoding rate. In MPEG-5 EVC, intra prediction consists in a set of 5 predefined linear functions where the mode yielding the best Rate-Distortion (RD) tradeoff is selected. However, not all contents (e.g., complex textures) can accurately be predicted by simple linear models, and in such cases the efficiency of intra prediction drops.

In this work we aim to improve intra prediction as specified by MPEG-5 EVC with a learnable predictor. We address the problem of predicting a block given its context as an image inpainting problem. Recently, deep convolutional generative neural networks have shown to outperform existing image inpainting methods thanks to their ability to learn highly non linear functions. Namely, masked convolutional neural networks have been recently proposed for image inpainting exploiting the a priori information on missing pixels that are weighted out from the context used to recover the missing image area. The method we propose relies on masked convolutions to generate the block predictor starting from the decoded context available at the receiver. In detail, we replaced the MPEG-5 EVC predictor mode 0 (i.e., the DC prediction) with a novel predictor that is computed by a masked convolutional autoencoder for each block to be encoded. Our encoding experiments in standard test conditions show Bjøntegaard Delta Rate (BD-Rate) reductions in excess of 6% over the MPEG-5 EVC Baseline Profile.

2 Background

This section first provides a primer to video coding, next reviews the state of the art in learnable intra-picture prediction.

2.1 Introduction to Video Coding

Existing video coding standards 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. In state of the art video coding standards such as the H.265/HEVC or MPEG-5 EVC, the image is first recursively subdivided in blocks (*Coding Units - CUs*) of decreasing size, e.g. 64×64 down to 4×4 in the MPEG-5 EVC standard. Next, for each coding unit multiple encoding modes are evaluated by an algorithm aimed at finding the best RD tradeoff for a given Quantization Parameter (QP). Better RD tradeoffs can be achieved by predicting the coding unit from neighboring data within the same picture (intra-prediction)

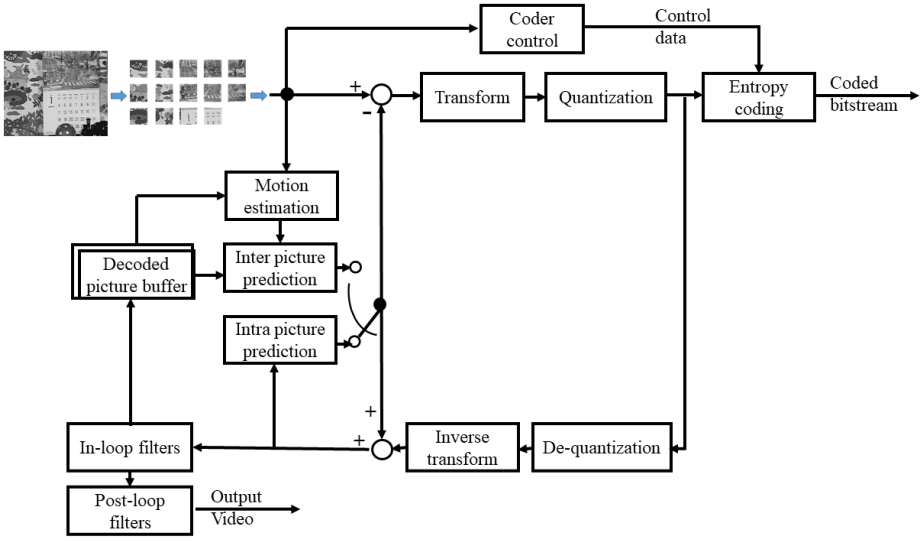

Fig. 1: Architecture of a traditional hybrid video codec with the main coding tools: this work deals with enhancing the intra tool with a learnable predictor.

or from previously encoded pictures if available (inter-prediction). Intra-frame prediction leverages the spatial correlation within the same picture generating a predictor for the CU to be encoded by extrapolating pixel values from a previously encoded neighborhood. The predicted block is then subtracted from the original block, producing a residual block that is transformed via discrete cosine transform, allowing low-pass filtering in the transformed domain by discarding and/or attenuating the coefficients in the high frequency values. 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 Baseline profile includes 5 intra prediction modes: DC, horizontal, vertical and two diagonal modes for each CU. The encoder selects the intra mode that minimizes the residual rate, which may be then put into competition with other modes. Coefficient decimation and the subsequent quantization is the lossy part of the compression process that reduces the high frequency rate while keeping the resulting artifacts bearable to the human observer. The resulting signal is entropy encoded, via for example, arithmetic coded, which is a lossless form of compression. Within the encoder, a decoding part is implemented and the signal is reconstructed through a dequantization and inverse transformation step. By adding the predicted signal, the input data is reconstructed. 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 pic-

tures. At the decoder side, the signaled predictor is generated from the decoded context and then the residual is decoded, added to the predictor, recovering the encoded block.

2.2 Learning-based intra prediction

It is not surprising that the recent advances in deep generative models, such as auto-encoders and generative adversarial networks have stimulated research towards applying these tools to image and video compression [1, 19, 15]. Auto-encoder architectures [7, 10], in particular, are especially effective to obtain compressed latent representations, by forcing the output to reproduce the input image through an information bottleneck whose dimensionality is much smaller than the original input space. Image compression methods based on auto-encoders have been shown to yield coding gains compared to legacy image codecs such as JPEG and JPEG 2000, and competitive results with more recent image compression algorithms such as BPG [2, 19, 20].

The work in [6] is one of the earliest proposing using a set of Neural Networks (NNs) for generating an intra prediction. Namely, they show that while for small sized blocks a fully connected NN gives best results, convolutional networks yield better predictors for large blocks with complex textures. Their experiments integrating a learnable predictor into H.265/HEVC show PSNR gains above 5% in some cases, depending on the content type and how the predictor is integrated into the codec. The authors attribute such gains to the improved ability to correctly predict complex textures.

The same authors propose an iterative approach to training a NN for intra prediction in [5]. First, a NN is trained on blocks and context extracted from a real partitioning of pictures as produced by the reference codec. Next, the NN is refined over the output of the same codec, yet this time the output includes the learnable intra predictor trained during the previous step. It is shown that this train-and-refine approach boosts further the performance of the learnable intra predictor with BD-rate reduction in excess of 4% BD-rate with H.265/HEVC and close to 2% for H.266/VVC.

The work in [9] tackles the same problem yet with a different approach that relies on recurrent NNs. Namely, they propose a recurrent architecture with three different spatial recurrent units that progressively generate predictor pixels by passing information exploiting the already encoded context. Beside MSE, they train their model keeping into account the Sum of Absolute Transformed Difference (SATD) as a proxy of the rate. They experimentally show that their approach yields bit rate reductions in excess of 2.5% when integrated into H.265/HEVC.

In [8], a NN that has multiple prediction modes and that co-adapts during training to minimize a loss function is proposed. The proposed loss function reflects the properties of the residual quantization of the typical hybrid video coding architecture by applying the ℓ_1-norm and a sigmoid-function to the prediction residual in the DCT domain. Furthermore, they reduce the complexity

by pruning the resulting predictors in the frequency domain and by quantizing the network weights and utilizing fixed point arithmetic, thus allowing for a hardware-friendly implementation.

In [22], a slightly different approach is proposed, where a NN is used to refine the standard H.265/HEVC intra prediction modes rather than replacing them. Such approach builds upon a convolutional autoencoder that is trained to recover a missing area of an image by inpainting the masked pixels corresponding to the block to be predicted. The authors in [22] experimentally show that their approach reduces up to 25% the mean square error of the H.265/HEVC intra predictor without additional signalling in the bitstream.

So far, no one has yet evaluated a learnable predictor within the MPEG-5 EVC codec. To the best of our knowledge, this is the first work evaluating to which extent a learnable intra predictor can affect the efficiency of a royalty free codec.

3 Proposed Method

In this section, we first describe the architecture of the NN we use to generate an intra predictor from a decoded context and next we detail the training procedure. Indeed, generating an intra prediction given the previously decoded context is conceptually equivalent to inpainting an image region given the available neighbor pixels. Therefore, we recast intra picture prediction as an image inpainting problem, building upon the existing body of research on the topic.
GS: Recently, image inpaiting models adopted mechanisms of attention or trasformers to capute long-range dependencies [11, 21, 23]. However, these models requires a large number of parameters and in some cases a minimum input size [11]. Since we have to work with small crops, however, we preferred to adopt a simpler convolutional network while keeping the number of parameters under control.

3.1 Network Architecture

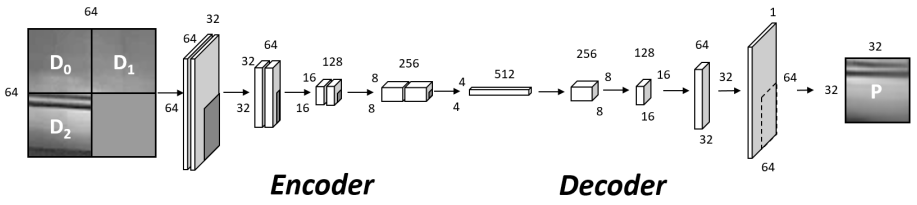

Fig. 2: Architecture and procedure for training the convolutional autoencoder used to generate a learnable intra predictor. In this example, a 32×32 prediction is generated from a 64×64 context.

Figure 2 shows the architecture of the convolutional autoencoder we propose to generate a predictor from a decoded context. For the sake of simplicity, we exemplify the case of a 32×32 predictor generated from a 64×64 context, however similar considerations hold for the other CU sizes supported by MPEG-5 EVC (16×16, 8×8, 4×4 CUs). The autoencoder receives as input a 64×64 patch representing the encoded context also available at the decoder (D_0, D_1, D_2) and outputs a 32×32 patch P corresponding to the intra predictor. The design is inspired by the context encoders for hole filling [14], yet with a number of significant differences and improvements tailored towards this task.

Concerning the *encoder*, it relies on masked convolutions where the convolution operator is constrained to valid input pixels (first layer) or features (following layers) [12]. In a nutshell, with masked convolutional layers learned filters operate only on pixels (or features) of the input image (or a feature map) that are not masked. For each convolutional layer, both a feature map and a binary mask are generated so that multiple masked layers can be stacked together.

Second, we stack pairs of masked convolutional layers with 3×3 filters and leaky ReLUs where the the filters of the first layer of the pair has 1 unit stride, whereas the second layer of the pair has stride of two and takes care of feature map downsampling replacing the pooling operator. We experimentally verified that this architecture reduces the number of learnable parameters as well as both the loss at training time and the intra-predictor rate at coding time.

Third, rather than projecting the input image on a 1×1 latent space, we project it on a vector of feature maps sized 4×4 by dropping one convolutional layer. Again, we experimentally verified that this setup yields both lower losses at training time and better efficiency at encoding time. We attribute such improvements to a spatial-semantic depth tradeoff that is more appropriate for the purpose of our task. This result is in line with recent learnable video codecs such as [1, 2], where the latent space encoded as bitstream is actually a serialization of a variable number of 4×4 feature maps.

The *encoder* is thus composed by 4 blocks of masked convolutions where each block is composed of two stacked masked convolutional layers as detailed in Figure 2. The output of the *encoder* is finally a collection of 512 feature maps sized 4×4 in the $[-1, 1]$ range.

Concerning the *decoder*, it is composed by a stack of 4 deconvolutional layers of size 4×4. Each deconvolutional layer doubles the resolution of feature maps in input, reversing the spatial subsampling performed at the *encoder*. Each deconvolutional layer is followed by a leaky ReLU activation, except the last that is followed by a hyperbolic tangent. The autoencoder output is finally a 64×64 image from where we crop the 32×32 intra predictor P in the figure where pixel values are in the $[-1, 1]$ range. We experimentally verified that while generating a 64×64 image for the purpose of cropping a patch is not strictly necessary, that improves both loss and encoder efficiency. Moreover, with a single network topology we can cope with CUs of different size (in our case, 16x16, 8x8 and 4x4) simply changing the crop operator geometry at the network output. Overall,

the autoencoder counts about 6M parameters, where about 4M are for learning convolutional filters and 2M for learning the convolution masks.

3.2 Training

The autoencoder is trained by minimizing the error between the predictor P and the original patch O on a dataset of about 1000 images of different resolution and content type randomly sampled from the AROD dataset [17]. While these images are JPEG compressed, they are very high quality, and so they cannot be told from uncompressed images. We found that training the autoencoder on high quality images is of pivotal importance, even when the trained autoencoder receives in input a context encoded at high QPs. We also found out that training on larger datasets such Imagenet, Vimeo or BVI-DVC did not provide significant advances despite longer training times.

From each image, a 64×64 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 network from overfitting on the training data. The bottom right 32×32 corner of the patch represents the original CU to recover, whereas the rest of the patch represents the (D_0, D_1, D_2) context. Prior to training, we prepare an appropriate binary mask that is provided in input to the first masked convolutional layer together with the context. The autoencoder is trained with SGD with a learning rate of 0.01 and over batches of 64 patches.
Ideally, the autoencoder shall be trained to minimize the linear combination of the rate and distortion terms corresponding to the operating point selected by the MPEG-5 EVC encoder [1, 2]. However, for the sake of simplicity, we follow the approach used in other similar works such as [6] where the network is trained at minimizing the reconstruction loss only. In the original context encoder [14], the network is trained to minimize a linear combination of L2 loss (i.e., the mean square error) and an adversarial term. The adversarial term was shown to produce sharper and more visually pleasant results than a L2 loss alone. However, we found that the adversarial term yields artifacts that albeit visually pleasant do not help reducing the residual rate. Most important, we found out that minimizing the L1 loss (i.e., the absolute error) yields smaller residuals and thus lower rates. We hypothesize that the L2 term gives much more weight to a few training samples that yield a high loss value yet do not represent the average case for the MPEG-5 EVC encoder.

3.3 Integration into the MPEG-5 EVC encoder

Once the autoencoder has been trained, it is interfaced with the MPEG-5 EVC encoder as follows. First, an external networked server process is started. The server loads the trained autoencoder into the GPU memory, sets up an UDP socket in listening mode and awaits for incoming messages. The EVC encoder is modified so that when an intra predictor has to be generated, the corresponding context D_1, D_2, D_3 is extracted from the currently encoded frame and is sent

to the server above over an UDP socket. The server inputs such context to the trained autoencoder and returns the 32×32 output P, i.e. the learned predictor, to the encoder again via the 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 MPEG-5 EVC encoder replaces the predictor with the autoencoder generated predictor and the encoding proceeds as usual, i.e., by putting the learned predictor in competition with other encoding modes.

Following the approach of [6], we consider two different approaches to integrate the trained autoencoder output within the MPEG-5 EVC intra prediction scheme.

The first approach consists in replacing the DC predictor (mode 0) with our learnable predictor for a total of 5 prediction modes. In [6] it is proposed to replace with the H.265/HEVC intra mode that is less likely to be selected due to the contextual intra mode signaling scheme H.265/HEVC employes. Conversely, in MPEG-5 EVC intra modes are simply signaled with variable length codes, so it is key that most probable modes are assigned shorted codes. Under the hypothesis that our learnable predictor is going to be picked by the RDO algorithm at least as frequently as the DC mode, we replaced the DC predictor with our learnable predictor.

The second approach consists in adding a sixth intra prediction mode for our learnable predictor aside the five MPEG-5 EVC intra modes. For the same reasons as above, we map our predictor to mode 0, whereas the DC predictor becomes mode 1, and so forth.

We point out that both the schemes above yield a completely decodable bitstream without the need for any side information under the reasonable assumption that the MPEG-5 EVC decoder has available the same autoencoder used by the encoder. Moreover, while the first approach is standard compliant as we do not change the bitstream, the latter only requires a simple modification at the decoder to parse the bitstream.

4 Experimental Results

4.1 Setup

We experiment encoding the first frame of the JVET CTC sequences at QP values in $[22, 27, 32, 37, 42]$ as recommended by MPEG for their experiments with the NNVC reference software [13]. Our learnable intra predictor is applied to CUs of size 32×32, 16×16, 8×8 and 4×4; only CUs 64×64 do not enjoy our learnable predictor as our experiments showed no appreciable marginal gains.

As a preliminary experiment, we visually inspect the generated predictors for four different contexts in Figure 3. The learnable predictor is able to inpaint the missing area of context with plausible predictions. With respect to the standard MPEG-5 EVC DC predictor, the learnable predictor yields better residual rates.

Table 1 shows the results of the encoding when our learnable predictor replaces the DC mode. The experiments report average BD-Rate improvements

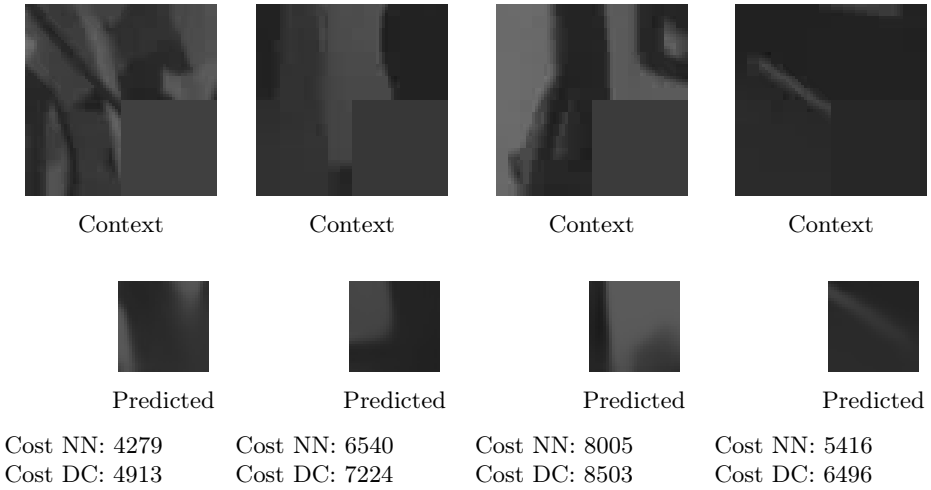

Fig. 3: Examples of 64×64 decoded context and 32×32 learnable predictor; the learnable predictor is capable of accounting also for complex texture patterns beyond simple linear interpolation. The reported "Cost" is the number of bits required to encode the residual.

in excess of 6% and BD-PSNR improvements in excess of 0.5 dB for some sequences. The experiments show gains especially for sequences with spatial resolution above 720p: a plausible explanation may stem from 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 video clips belonging to Classes C and D. The lowest performance is achieved for screen content (Class F), a non-unexpected result if we consider that text areas are more difficult to predict and our training set contains no computer screen images. A visual inspection of the decoded sequences shows no perceivable artefacts despite the learned intra predictor.

To gain a better understanding of these results, we performed a statistical analysis of the logs of the modified MPEG-5 EVC encoder for all the sequences in the table above. Table 2 shows the percentage of selection of each intra mode for Class A JVET sequences. With the reference MPEG-5 EVC encoder (left), the DC mode is selected about 51% of the times over the other 4 modes. When the DC predictor is replaced with our learnable predictor (center), this number increases to 62%, showing the advantage of a learnable predictor. That is, the intra mode indexed with code 0 is more likely to be signaled in the bitstream, and since it has the shortest code associated, the cost of intra signaling is reduced. Similarly, the residual rate for the learnable predictor was on average 9% lower than the equivalent rate of the EVC DC predictor. That is, replacing the DC predictor with our learnable predictor yields both lower residual and signaling rates if it is allocated mode 0, which explains the gains in Table 1.

Class	Sequence	BD-Rate	BD PSNR
Class A 3840x2160 60/50 fps 10 bpp	Campfire	-2.96	0.11
	CatRobot	-7.6	0.23
	DaylightRoad2	-8.03	0.2
	FoodMarket4	-10.09	0.23
	ParkRunning3	-1.94	0.13
	Tango2	-7.96	0.13
	Average	**-6.43**	**0.17**
Class B 1920x1080 60/50 fps 10/8 bpp	BQTerrace	-5.44	0.38
	BasketballDrive	-9.73	0.27
	Cactus	-6.87	0.29
	MarketPlace	-5.69	0.21
	RitualDance	-11.85	0.67
	Average	**-7.92**	**0.36**
Class C 832x480 60/50/30 fps 8 bpp	BQMall	-5.39	0.36
	BasketballDrill	-7.52	0.41
	PartyScene	-2.99	0.26
	RaceHorsesC	-6.03	0.45
	Average	**-5.48**	**0.37**
Class D 416x240 60/50/30 fps 8 bpp	BQSquare	-2.06	0.19
	BasketballPass	-4.20	0.27
	BlowingBubbles	-4.06	0.28
	RaceHorsesD	-5.21	0.42
	Average	**-3.88**	**0.32**
Class E 1280x720 60 fps 8 bpp	FourPeople	-12.82	0.83
	Johnny	-12.50	0.58
	KristenAndSara	-11.11	0.64
	Average	**-12.14**	**0.68**
Class F Screen content 60 fps 8 bpp	ArenaOfValor	-5.14	0.33
	BasketballDrillText	-6.09	0.35
	SlideEditing	-1.17	0.18
	SlideShow	-1.44	0.18
	Average	**-3.46**	**0.24**
	Grand Average	**-6.55**	**0.36**

Table 1: BD-Rate and BD-PSNR for MPEG5-EVC baseline profile integrated with our learnable intra predictor with respect to standard MPEG5-EVC baseline profile.

However, the analysis of the logs also revealed that the residual of the learnable predictor was lower than the residual of the DC predictor only in 53% of the cases. That is, in a significant number of cases the DC predictor is still a better predictor than the learnable predictor and replacing this latter with the learnable predictor is suboptimal in terms of residual costs. For this reason, we added a sixth mode for our learnable predictor, encoded as mode 0, whereas the DC predictor was mapped to mode 1, and so on. In this scenario, the learnable predictor is put into competition with the 5 standard EVC intra prediction

modes. We repeated the encodings and found that the overall BD-Rate and BD-PSNR improved only by 0.01 with respect to the numbers in Table 1. When the learnable predictor is put in competition with the other 5 modes (Table 2, right), it is selected only 56% of the times. We hypothesize that the 6-modes signaling rate leads to lower residual rates. However, the extra rate required for signaling the 6th mode counterbalances these gains, making this scheme less competitive than DC replacement in practice.

Table 2: Percentage of intra modes selection for JVET Class A sequences. Left: 5 modes, reference. Center: 5 modes, proposed. Right: 6 modes, proposed.

Mode	**%**	**Mode**	**%**	**Mode**	**%**
-	-	-	-	0 NN	56.0
0 DC	51.0	0 NN	62.0	1 DC	25.0
1 H	22.0	1 H	19.0	2 H	9.7
2 V	20.0	2 V	14.0	3 V	6.1
3 D1	4.7	3 D1	3.1	4 D1	1.8
4 D2	2.5	4 D2	1.6	5 D2	1.4

5 Conclusions and Future Works

We designed, trained and evaluated a learnable intra-picture predictor for a video codec compliant with the royalty free MPEG-5 EVC standard. Our experiments on standard test sequences show average BD-Rate gains in excess of 6% by replacing the standard DC predictor with our learnable predictor. When put into competition with the DC mode as an additional intra mode, our predictor still exhibits lower residual cost, however without appreciable gains in RD terms: we hypothesize that this is due to the increased signaling costs. Current endeavours of the MPAI EVC working group include enhancing the inloop filter with a learnable approach and resorting to a upsampling scheme outside the encoding loop.

References

1. Ballé, J., Laparra, V., Simoncelli, E.P.: End-to-end optimized image compression. In: Int. Conf. on Learning Representations (ICLR). Toulon, France (Apr 2017)
2. Ballé, J., Minnen, D., Singh, S., Hwang, S.J., Johnston, N.: Variational image compression with a scale hyperprior. In: Int. Conf. on Learning Representations (ICLR). Vancouver, CA (May 2018)
3. Choi, K., Chen, J., Rusanovskyy, D., Choi, K.P., Jang, E.S.: An overview of the mpeg-5 essential video coding standard [standards in a nutshell]. IEEE Signal Processing Magazine **37**(3), 160–167 (2020)

4. CISCO: Global 2021 forecast highlights. https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_2021_Forecast_Highlights.pdf (2021)
5. Dumas, T., Galpin, F., Bordes, P.: Iterative training of neural networks for intra prediction. IEEE Transactions on Image Processing **30**, 697–711 (2020)
6. Dumas, T., Roumy, A., Guillemot, C.: Context-adaptive neural network-based prediction for image compression. IEEE Transactions on Image Processing **29**, 679–693 (2019)
7. Goodfellow, I., Courville, A., Bengio, Y.: Deep learning, vol. 1. MIT press Cambridge (2016)
8. Helle, P., Pfaff, J., Schäfer, M., Rischke, R., Schwarz, H., Marpe, D., Wiegand, T.: Intra picture prediction for video coding with neural networks. In: 2019 Data Compression Conference (DCC). pp. 448–457. IEEE (2019)
9. Hu, Y., Yang, W., Li, M., Liu, J.: Progressive spatial recurrent neural network for intra prediction. IEEE Transactions on Multimedia **21**(12), 3024–3037 (2019)
10. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: Int. Conf. on Learning Representations (ICLR). Banff, CA (Apr 2014)
11. Li, W., Lin, Z., Zhou, K., Qi, L., Wang, Y., Jia, J.: Mat: Mask-aware transformer for large hole image inpainting. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10748–10758 (2022). https://doi.org/10.1109/CVPR52688.2022.01049
12. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: Proceedings of the European conference on computer vision (ECCV). pp. 85–100 (2018)
13. MPEG: Jvet common test conditions and evaluation procedures for neural network-based video coding technology. Output document of JVET (April 2023)
14. Pathak, D., Krähenbühl, P., Donahue, J., Darrell, T., Efros, A.: Context encoders: Feature learning by inpainting (2016)
15. Rippel, O., Bourdev, L.: Real-time adaptive image compression. arXiv preprint arXiv:1705.05823 (2017)
16. Samuelsson, J., Choi, K., Chen, J., Rusanovskyy, D.: Mpeg-5 evc. In: SMPTE 2019. pp. 1–11. SMPTE (2019)
17. Schwarz, K., Wieschollek, P., Lensch, H.P.: Will people like your image? learning the aesthetic space. In: 2018 IEEE winter conference on applications of computer vision (WACV). pp. 2048–2057. IEEE (2018)
18. Sze, V., Budagavi, M., Sullivan, G.J.: High efficiency video coding (HEVC). Integrated Circuit and Systems, Algorithms and Architectures. Springer **39**, 40 (2014)
19. Toderici, G., Vincent, D., Johnston, N., Hwang, S.J., Minnen, D., Shor, J., Covell, M.: Full resolution image compression with recurrent neural networks. In: IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 5435–5443. Honolulu, Hawaii, USA (Jul 2017)
20. Valenzise, G., Purica, A., Hulusic, V., Cagnazzo, M.: Quality Assessment of Deep-Learning-Based Image Compression. In: Multimedia Signal Processing. Vancouver, Canada (Aug 2018), https://hal.archives-ouvertes.fr/hal-01819588
21. Wan, Z., Zhang, J., Chen, D., Liao, J.: High-fidelity pluralistic image completion with transformers. CoRR **abs/2103.14031** (2021), https://arxiv.org/abs/2103.14031
22. Wang, L., Fiandrotti, A., Purica, A., Valenzise, G., Cagnazzo, M.: 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 (2019)

23. Yu, Y., Zhan, F., Wu, R., Pan, J., Cui, K., Lu, S., Ma, F., Xie, X., Miao, C.: Diverse image inpainting with bidirectional and autoregressive transformers. CoRR **abs/2104.12335** (2021), https://arxiv.org/abs/2104.12335