A Study of Deep-Learning-based Prediction Methods for Lossless Coding

Ionut Schiopu, Hongyue Huang, and Adrian Munteanu
Department of Electronics and Informatics
Vrije Universiteit Brussel
Brussels, Belgium
{ischiopu, hhuang, acmuntea}@etrovub.be

***Abstract*—In recent years, the research community started to explore new strategies for encoding image and video content based on innovative coding solutions developed using machine learning (ML) tools. An emerging research strategy proposes novel hybrid coding solutions as an alternative to traditional coding paradigms by replacing specific coding modules with efficient deep-learning (DL) based techniques. The paper presents a study on DL-based intra-prediction methods for lossless compression applications. For image coding, the paper studies our recently proposed pixel-wise prediction methods designed based on the residual learning concept, integrated into conventional lossless image coding frameworks. Moreover, a novel neural network design is proposed based on a new structure of layers. For video coding, the paper studies our recently proposed block-wise prediction methods designed based on recent breakthroughs in the ML domain, and integrated in the lossless HEVC standard. Experimental results show that the proposed lossless image codec achieves an improved performance with 1.6% compared to state-of-the-art DL-based methods. The study reveals that the hybrid coding solutions which incorporate DL-based prediction methods systematically and substantially improve the coding performance over traditional lossless coding paradigms.**

***Index Terms*—Lossless compression, deep-learning, intra-prediction, hybrid coding solutions**

I. Introduction

Machine Learning (ML) based technologies already provide an improved performance compared to traditional state-of-the-art methods in numerous image processing applications. In the data compression domain, a novel research strategy emerged by proposing an innovative hybrid coding solution as an alternative to the traditional coding solutions by training different neural network models to replace specific task-oriented modules in conventional coding frameworks, such as CALIC [1] and High Efficiency Video Coding (HEVC) [2].

In recent years, several approaches were proposed to integrate modern ML tools into lossy coding systems. In [3], the authors proposed one of the first image compression methods that employs modern ML tools. In [4], an end-to-end trainable model for image compression based on variational auto-encoders is proposed, where the model incorporates a hyperprior to effectively capture spatial dependencies in the latent representation. A variable-filter-size residue-learning convolutional neural network model was proposed in [5], providing an accelerated network training and improving the HEVC intra-coding performance. In [6], the authors proposed an arithmetic coding strategy by directly estimating the probability distribution of the 35 intra-prediction modes with the adoption of a multi-level arithmetic codec, and by training a simple neural network to perform probability estimation. In [7], the authors proposed a neural network design based on a sequence of several dense layers which is employed for block-based intra prediction in lossy video coding. In [8], a set of features from the causal neighborhood is extracted and used in order to select a predefined image pattern as the prediction signal.

In the lossless coding domain, several approaches were proposed to compute an improved deep-learning (DL) based prediction, where the residual error is further encoded using traditional coding tools. For lossless image coding applications, several pixel-wise DL-based approaches were proposed to compute the prediction of the current pixel by classifying the causal neighborhood of the current pixel into a predefined number of classes. In [9], we proposed a first solution for lossless coding of photographic images. In [10], a dual prediction method based on traditional and DL-based prediction is proposed. In [11], we proposed a network design using ML concepts such as Inception [12] and Residual Learning [13] and a novel bit-plane-based entropy codec. For lossless video coding applications, several block-based DL-based approaches are proposed to compute the current block prediction using different strategies for selecting the causal neighborhood. In [14], we proposed a novel DL-based architecture and trained specific network models in order to replace conventional HEVC predictions for nine intra-prediction modes for the 4×4 block size. Moreover, in [15], we proposed a hybrid coding solution based on an improved training procedure where the optimal intra-prediction mode is selected between the angular prediction modes of both HEVC-based and DL-based approaches, for the 4×4 and 8×8 block sizes.

The paper studies various ML-based intra prediction methods for lossless coding applications and assesses their coding performance compared to traditional coding methods. For lossless image coding, the paper proposes a novel neural network design based on a more efficient 3-branch structure of dense layers and studies different architecture variations. Section II

This work was supported by Agentschap Innoveren en Ondernemen (VLAIO) within the research project imec icon ILLUMINATE HBC.2018.0201, and by the Fonds Wetenschappelijk Onderzoek (FWO) - Vlaanderen within the research project G084117.

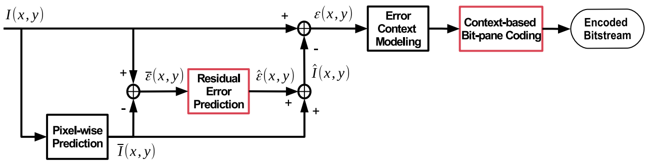

Fig. 1. The hybrid lossless image coding solution. The red rectangles mark the modules improved in [11].

studies the DL-based intra prediction methods. Section III analyses the performance of DL-based intra prediction for image and video coding. Section IV concludes this study.

II. Deep-Learning-based Intra-Prediction

A. Pixel-wise prediction for image coding

The conventional lossless coding framework was modified by integrating a novel dual prediction scheme [10] and an efficient entropy coder [11]. Fig. 1 depicts the hybrid coding solution proposed in [11], where: **(1)** a traditional prediction method is first employed to compute the initial pixel prediction based on a small causal neighborhood; **(2)** a DL-based method designed based on the inception and residual learning concepts is then employed to predict the residual error using a large causal neighborhood, and **(3)** a context-based bit-plane (CBP) codec is employed to encode the final residual error.

The traditional state-of-the-art methods for lossless compression were conceived years ago on low resolution imagery by respecting severe design constraints on computational complexity. E.g., the LOCO-I [16] and GAP [1] prediction methods follow the predictive coding paradigma whereby the value of the current pixel is predicted using a linear combination of the values in a small causal neighborhood of three-to-six pixels. The DL-based methods propose to estimate the value of the current pixel based on a rectangular input patch of size $(b+1) \times (2b+1)$ which collects the known pixel values found in a causal neighbourhood at a maximum of b lines or columns from the current pixel position. DL-based methods follow the idea that the context provided by a larger causal neighborhood can be used to efficiently classify the input patch into a class of which the index represents the predicted value. One notes that, in contrast to traditional lossless codecs, DL-based coding methods were conceived to maximize lossless coding performance without any design constraint on computational complexity.

1) Related Work: In our recent work, we proposed the Residual-Error Predictive Convolutional Neural Network (REP-CNN) [10] and the Inception and Residual Learning-based Neural Network (IResLNN) [11] architectures depicted in Fig. 2f using the different structures of layers detailed in Figs. 2a-2d. These structures are placing a batch normalization (BN) [17] layer between the activation function and a convolution (Conv) or a dense (Dense) layer. The goal of these neural network designs is to classify the input patch of local causal neighborhood into 256 classes using a $2-$stage approach. In the first stage, a structure based on *convolution* layers process the input patch to extract a feature vector. In the second stage, a structure based on *dense* layers further process the feature vector to compute the class index.

REP-CNN [10] consists of a sequence of 14 Convolution Blocks (CBs) in the first stage, and 12 Dense Blocks (DBs) in the second stage, where the last DB block uses a *softmax* activation function to classify the input patch [11]. IResLNN [11] proposes to change the the first stage of REP-CNN by introducing two efficient structures of convolution layers: the Residual Learning [13] based Block (ResLB), and the Inception [12] and Residual Learning based Block (IResLB). These structures are using a multiple branch processing design, where each branch processes the current patch in a different way. Hence, IResLNN proposes a more elaborated design consisting of a sequence of 2 CBs, 5 IResLBs, and 2 ResLBs.

The residual errors are truncated from a $9-$bit to a $8-$bit dynamic range (with 256 values) based on the observation that large-magnitude residuals are highly unlikely to occur. The training procedure employs the Adam optimizer [18], and a loss function based on sparse categorical cross-entropy and by using ℓ_2 regularization with hyperparameter $\lambda = 0.01$.

2) Proposed Network Designs: Here we propose a new network design based on a novel dense layer structure called Dense IResLB (DIResLB), depicted in Fig. 2e, which proposes to apply the multiple branch processing approach using dense layers to feature classification. DIResLB uses parameter d to variate the number of classes in *branch 2* and *branch 3*, so that in each branch the patch is processed in a different way.

Firstly, we propose to modify the second stage in IResLNN by replacing the sequence of 11 DBs with two DIResLB blocks. Five different ways to process the feature vector are proposed. Fig. 2f depicts the full architecture design of the first architecture variation, denoted *Proposed3*, where the two DIResLB blocks use $d = 64$. Fig. 2g depicts the other strategies tested for replacing the layer structure marked by the dotted rectangle in *Proposed3*. A second version, denoted *2 DIResLB(0)*, proposes to test the DIResLB block with $d = 0$. A third version, denoted *2 DIResLB(96)*, proposes to test the DIResLB block with $d = 96$ and introduce a higher parameter variation between the branches. A fourth version, denoted *3 DIResLB(64)*, proposes to increase the number of parameters by using 3 DIResLBs. A fifth version, denoted *5-b DIResLB*, proposes to modify the DIResLB structure by using a $5-$branch design, as depicted in Fig. 2g.

Secondly, we propose to further modify the first stage in the *Proposed3* architecture by changing the IResLB structure and use a 5×5 window instead of a 3×3 window for Conv_2, see Fig. 2d. The obtained architecture is denoted *Proposed5*.

3) Entropy Coding: A new entropy codec was proposed to encode the residual error of the dual prediction method using its binary representation $\sum_{i=0}^{i=k_\epsilon(x,y)} b_i 2^i$, where $b_i \in \{0,1\}$ and $k_\epsilon(x,y) = 0, 1, \ldots, 7$ is the (minimum) number of bits needed to represent the residual error. The length $k_\epsilon(x,y)$ is first predicted based on a small causal neighborhood and each bit in the binary representation is then predicted by employing a different context tree model [19]. The CBPNN method

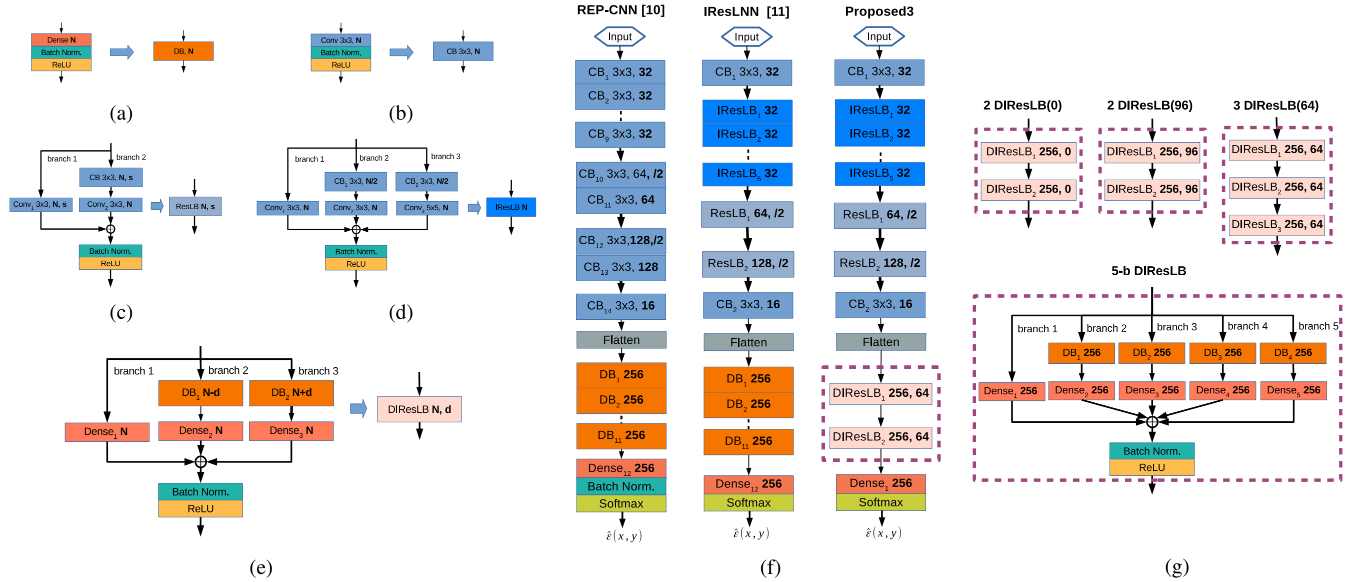

Fig. 2. Layer structure of the blocks from [11]: **(a)** Dense Block (DB); **(b)** Convolution Block (CB); **(c)** Residual Learning based Block (ResLB); **(d)** Inception and ResLB (IResLB). Proposed layer structure: **(e)** Dense IResLB (DIResLB). **(f)** The neural network design of REP-CNN [10] and IRESLNN [11], and *Proposed3* architecture. **(g)** The *2 DIResLB (0)*, *2 DIResLB(96)*, *3 DIResLB(64)*, and *5-b DIResLB* variations.

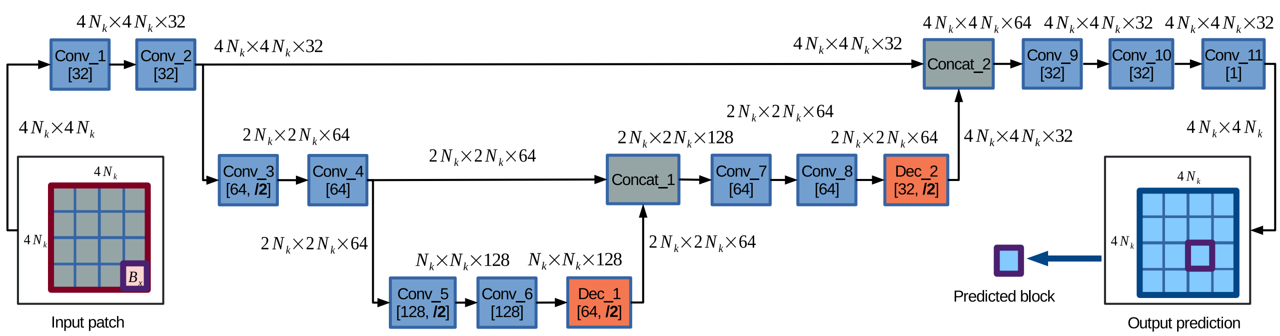

Fig. 3. The network design first proposed in [14] and further modifying in [15].

was proposed by applying a dual prediction method based on LOCO-I [16] and IResLNN for predicting the LOCO-I residual error, and by encoding the final residual using CBP.

B. Block-wise prediction for video coding

In the HEVC framework [2], a block-wise prediction method is employed to compute the Prediction Units (PU) of blocks whose size varies from 4×4 to 64×64 for each of the 35 HEVC intra-prediction modes. In our coding strategy we propose to employ a block-wise DL-based prediction method to compute the prediction of the current block based on an input patch having four times the size of the target prediction block, and which collects a causal neighborhood around the currently predicted block. Such approach is based on neural network designs that follow a multi-resolution processing concept, whereby the current patch processing is further improved based on the patches processed at the current resolution level and at half of the current resolution.

In [14], we proposed the Angular Intra-Prediction Convolutional Neural Network (AP-CNN) where the input patch is processed at three different resolutions. An AP-CNN model was trained to replace a set of nine HEVC angular intra-prediction modes for the prediction blocks of size 4×4, i.e., the In the lossless video coding domain, the prediction module plays an important role, and that the 4×4 block size was found as the most frequent block size in the optimal HEVC mode segmentation. Moreover, these nine angular intra-prediction modes were selected due to the large volume of input patches available for training. AP-CNN operates on three resolutions ($N_k \times N_k$, $\frac{N_k}{2} \times \frac{N_k}{2}$, $\frac{N_k}{4} \times \frac{N_k}{4}$) and contains 10 convolutional layers and 2 deconvolution layers. The number of filters is doubled when the patch resolution is halved, and the number of filters is halved when the patch resolution is doubled. A final convolution layer computes the final prediction block.

In [15], we proposed a hybrid video coding solution where, for the block prediction of the $N_1 \times N_1 = 4 \times 4$ ($k = 1$) and $N_2 \times N_2 = 8 \times 8$ ($k = 2$) block sizes, the optimal intra-prediction mode is selected between the following intra-prediction modes: the DC mode, the planar mode, the 33 traditional HEVC angular intra-prediction modes, and the 33

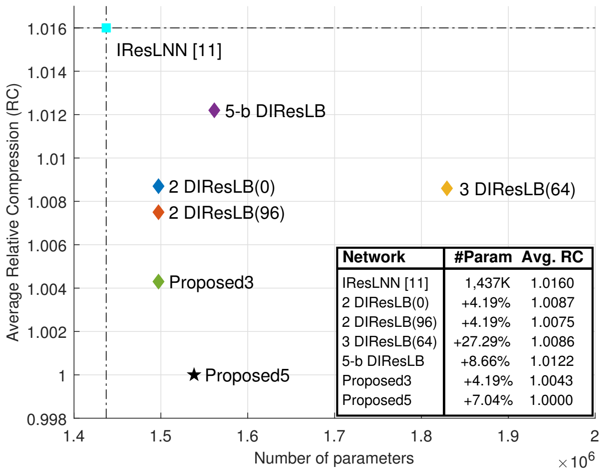

Fig. 4. Complexity vs. performance study.

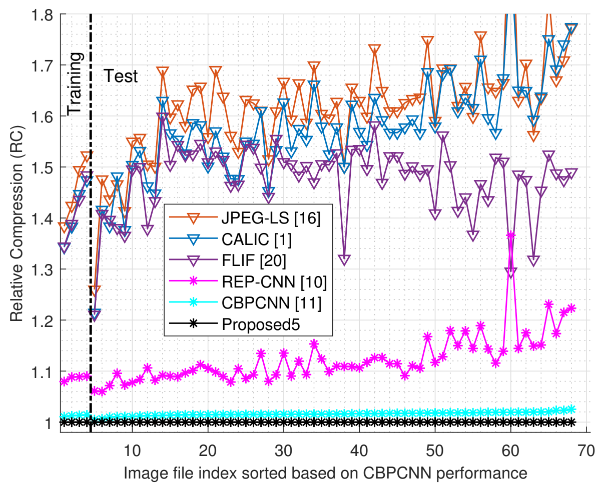

Fig. 5. Relative compression results for UHD images.

DL-based angular intra-prediction modes. Fig. 3 shows how AP-CNN was further modified in [11] based on the following improved training procedure. Firstly the neural network is trained to learn more efficiently from both the input patch and the target output prediction block, i.e., by predicting an area larger than the target prediction block. Secondly, the loss function formulation is based on a general loss term, a local loss term, and a rate-based term where the position of each predicted value relative to the causal neighborhood is taken into account for an improved block-wise intra-prediction. Thirdly, the input training patches are extracted after processing regular images rather than video sequences.

Furthermore, in [15], the HEVC video coding standard was modified to apply a block rotation to cancel the effect of the reverse raster scan introduced to collect the transformed coefficients in an order from low to high frequencies. This modification is based on the observation that a block-based prediction method has an improved performance for the top-left samples compared to the bottom-right samples since they are highly correlated with the causal neighborhood.

TABLE I
LOSSLESS IMAGE COMPRESSION RESULTS

Avg.	Traditional codecs			Hybrid coding solutions		
	JPEG-LS [16]	CALIC [1]	FLIF [20]	REP-CNN [10]	CBPNN [11]	Proposed5 codec
bpp	3.055	2.964	2.831	2.134	1.952	**1.923**
RC	1.618	1.573	1.474	1.123	1.016	**1.000**

III. EXPERIMENTAL RESULTS

A. Lossless Image Compression

The set of 68 grayscale 4K UHD photographic images [21] was divided into Training set (4 images) and Test set (64 images). Each model is trained using 10 million (M) input patches of size 16×31 ($b = 15$), which were randomly selected from the Training set, during 32 epochs, with batch size of 4000 input patches, and using a $90\% - 10\%$ training$-$validation ratio. Here we introduce a two stage training procedure for training the proposed architecture variation. In the first stage, we employ the procedure from [11] where at each epoch i, the learning rate η_i is set as $\eta_{i+1} = (f_d)^{\lfloor \frac{i}{n_s} \rfloor}\eta_i, \forall i = 1, 2, \ldots, 32$, where $f_d = 0.2$ is the decay rate, $n_s = 5$ is the decay step, and $\eta_1 = 5 \times 10^{-4}$ is the learning rate at the first epoch. In the second stage, the model trained in the first stage is further trained during $n' = 32$ epochs using a decay rate of $f'_d = 0.5$, a decay step of $n'_s = 5$, and a learning rate at first epoch of $\eta'_1 = 10^{-5}$.

The LOCO-I [16] predictor is employed in the dual prediction scheme [10], the LOCO-I residual error is then predicted using a different network model, and the final residual error is encoded by CBP. The compression results are compared using the bits per pixel (bpp) and the relative compression (RC) metrics, where $RC_{\text{MX}} = \frac{BR_{\text{MX}}}{BR_{\text{Proposed5}}}$ is the bitrate ratio between a codec MX and the tested anchor codec based on the *Proposed5* architecture.

Fig. 4 shows the complexity vs. performance analysis of IResLNN [11] and the proposed architecture variations: (i) *2 DIResLB(0)*, (ii) *2 DIResLB(96)*, (iii) *3 DIResLB(64)*, (iv) *5-b DIResLB*, (v) *Proposed3*, (vi) *Proposed5*. One can note that the *Proposed5* architecture achieves 1.6% average performance improvement compared to IResLNN [11]. Moreover, all the proposed architecture variations based on the more efficient DIResLB structure achieve an improved performance compared to IResLNN [11].

Fig. 5 shows the comparison between the compression results of the traditional lossless image codecs: JPEG-LS [16], CALIC [1], FLIF [20], and the novel hybrid lossless image coding solutions: REP-CNN [10], CBPCNN [11] (IResLNN prediction and CBP), and the Proposed5 codec (*Proposed5* prediction and CBP). Table I shows that the proposed codec based on the novel network design achieves an improved average performance of almost 50% compared to FLIF [20].

B. Lossless Video Coding

In [15], $10M$ and $2.5M$ input patches where extracted from the ADE20K dataset [22] for the 4×4 and 8×8 block sizes,

TABLE II
LOSSLESS VIDEO CODING RESULTS

Video Sequence	HEVC [2]	Hybrid Method [15] *bpp*	Hybrid Method [15] Δ_{BR}	Hybrid Method [15] Avg.Δ_{BR}	AP-CNN [14] Avg.Δ_{BR}
Traffic	3.927	3.606	**-8.169%**	**-8.660%**	-1.778%
PeopleOnStreet	4.038	3.668	**-9.150%**		
Kimono	3.729	3.553	**-4.724%**	**-3.561%**	-0.932%
ParkScene	4.452	4.237	**-4.836%**		
Cactus	4.461	4.321	**-3.122%**		
BQTerrace	4.726	4.597	**-2.731%**		
BasketballDrive	4.096	3.998	**-2.391%**		
RaceHorses	4.357	4.160	**-4.524%**	**-3.725%**	-0.289%
BQMall	4.284	4.112	**-4.006%**		
PartyScene	5.433	5.263	**-3.134%**		
BasketballDrill	4.157	4.023	**-3.235%**		
RaceHorses	4.638	4.329	**-6.655%**	**-4.637%**	-0.404%
BQSquare	5.490	5.344	**-2.651%**		
BlowingBubbles	5.512	5.321	**-3.463%**		
BasketballPass	4.165	3.925	**-5.778%**		
Overall Improvement				**-5.146%**	-0.851%

Fig. 6. Pseudo-colored images from [15]: (left) *Traffic* sequence, (right) *ParkScene* sequence. Green, red, and blue marks the blocks where the DL-based mode has a better performance, the HEVC-based mode has a better performance, and both modes have same performance, respectively.

respectively. The Adam Optimizer [18] was employed with a batch size of 500 input patches. Each model was trained during 20 epochs and the same $90\% - 10\%$ ratio was used to split the data into training−validation data.

Table II shows the results for lossless HEVC [2] under All-Intra profile, denoted here HEVC, AP-CNN [14], and Hybrid Method [15], for 15 videos from the Common-Test dataset [23], having one of the following frame resolutions: 416×240, 832×480, 1920×1080, 2560×1600. The bitrate improvement, Δ_{BR}, of a method MX compared to HEVC [2] is computed as $\Delta_{BR} = \frac{BR_{\text{MX}}}{BR_{\text{HEVC}}} - 1$. Table II shows that Hybrid Method [15] achieves around 5.1% improvement compared to HEVC.

Fig. 6 shows the pseudo-colored images generated for the pair of Hybrid Method [15] and HEVC [2] codecs. The large areas colored in green prove that DL-based methods are able to improve intra-prediction for a large number of prediction blocks, leading to notable coding gains compared to HEVC.

IV. CONCLUSIONS

The proposed study analyzes the performance of DL-based intra-prediction methods for lossless coding applications. The results demonstrate a systematic and substantial improvement of the lossless coding performance compared to conventional lossless coding methods. These results demonstrate that traditional methods can be successfully replaced by efficient DL-based methods for intra-prediction. Based on the outstanding performance gap achieved by the hybrid coding solutions compared to the traditional methods, we conclude that opting for DL-based intra coding methods is advisable in future lossless image and video coding standards.

REFERENCES

[1] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," IEEE Trans. Commun., vol. 45, no. 4, pp. 437–444, Apr. 1997.

[2] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[3] G. Toderici et al., "Full resolution image compression with recurrent neural networks,“ IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, 2017, pp. 5435–5443.

[4] J. Balle, D. Minnen, S. Singh, S. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior,“ Int. Conf. Learning Representations, Vancouver, BC, Canada, 2018, pp. 1–47.

[5] Y. Day, D. Liu, and F. Wu, "A convolutional neural network approach for post-processing in HEVC intra coding," Int. Conf. MultiMedia Modeling, Reykjavik, Iceland, 2017, pp. 28–39.

[6] R. Song, D. Liu, H. Li, and F. Wu, "Neural network-based arithmetic coding of intra prediction modes in HEVC," IEEE Visual Communications and Image Processing, St. Petersburg, FL, 2017, pp. 1-4.

[7] J. Li, B. Li, J. Xu, R. Xiong, and W. Gao, "Fully connected network-based intra prediction for image coding," IEEE Trans. Image Process., vol. 27, no. 7, pp. 3236-3247, July 2018.

[8] J. Pfaff et al., "Neural network based intra prediction for video coding," Proc. SPIE, vol. 10752, pp. 13–1–8, Sep. 2018.

[9] I. Schiopu, Y. Liu, and A. Munteanu, "CNN-based prediction for lossless coding of photographic images," Picture Coding Symposium, San Francisco, CA, 2018, pp. 16-20.

[10] I. Schiopu and A. Munteanu, "Residual-error prediction based on deep learning for lossless image compression," IET Electronics Letters, vol. 54, no. 17, pp. 1032-1034, 23 8 2018.

[11] I. Schiopu and A. Munteanu, "Deep-learning based lossless image coding," IEEE Trans. Circuits Syst. Video Technol. (Early Access).

[12] C. Szegedy et al., "Going deeper with convolutions," Conference on Computer Vision and Pattern Recognition, Boston, MA, 2015, pp. 1-9.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, 2016, pp. 770-778.

[14] H. Huang, I. Schiopu, and A. Munteanu, "Deep learning based angular intra-prediction for lossless HEVC video coding," Data Compression Conference, Snowbird, UT, 2019, pp. 579–579.

[15] I. Schiopu, H. Huang, and A. Munteanu, "CNN-based intra-prediction for lossless HEVC," IEEE Trans. Circuits Syst. Video Technol. (Early Access).

[16] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," IEEE Trans. Image Process, vol. 9, no. 8, pp. 1309-1324, Aug. 2000.

[17] S. Ioffe and C Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," Int. Conf. Machine Learning, Lille, France, 2015, pp. 1–9.

[18] D.P. Kingma and L.J. Ba, "Adam: A method for stochastic optimization," Int. Conf. Learning Representations, San Diego, CA, 2015.

[19] I. Schiopu, "Depth-map image compression based on region and contour modeling." Tampere University of Technology. Publication, Vuosikerta. 1360, Tampere University of Technology, 1 2016.

[20] J. Sneyers and P. Wuille, "FLIF: Free lossless image format based on MANIAC compression," IEEE Int. Conf. Image Processing, Phoenix, AZ, 2016, pp. 66-70.

[21] "UHD Photographic Images," http://www.ultrahdwallpapers.net/nature, Accessed: 2017-08-25.

[22] B. Zhou et al., "ADE20K Dataset," groups.csail.mit.edu/vision/datasets/ADE20K, Accessed: 2018-12-20.

[23] F. Bossen, "Common HM test conditions and software reference configurations," JCT-VC document, JCTVC-G1100, San Jose, CA, 2012.