Image and Video Compression With Neural Networks: A Review

Siwei Ma, *Member, IEEE*, Xinfeng Zhang, *Member, IEEE*, Chuanmin Jia, *Student Member, IEEE*, Zhenghui Zhao, Shiqi Wang, *Member, IEEE*, and Shanshe Wang

Abstract—In recent years, the image and video coding technologies have advanced by leaps and bounds. However, due to the popularization of image and video acquisition devices, the growth rate of image and video data is far beyond the improvement of the compression ratio. In particular, it has been widely recognized that there are increasing challenges of pursuing further coding performance improvement within the traditional hybrid coding framework. Deep convolution neural network which makes the neural network resurge in recent years and has achieved great success in both artificial intelligent and signal processing fields, also provides a novel and promising solution for image and video compression. In this paper, we provide a systematic, comprehensive and up-to-date review of neural network-based image and video compression techniques. The evolution and development of neural network-based compression methodologies are introduced for images and video respectively. More specifically, the cutting-edge video coding techniques by leveraging deep learning and HEVC framework are presented and discussed, which promote the state-of-the-art video coding performance substantially. Moreover, the end-to-end image and video coding frameworks based on neural networks are also reviewed, revealing interesting explorations on next generation image and video coding frameworks/standards. The most significant research works on the image and video coding related topics using neural networks are highlighted, and future trends are also envisioned. In particular, the joint compression on semantic and visual information is tentatively explored to formulate high efficiency signal representation structure for both human vision and machine vision, which are the two dominant signal receptors in the age of artificial intelligence.

Index Terms—Neural network, deep learning, CNN, image compression, video coding.

S. Ma, C. Jia, Z. Zhao, and S. Wang are with the Institute of Digital Media, School of Electronic Engineering and Computer Science, Peking University, Beijing 100871, China, and also with the Peng Cheng Lab, Shenzhen 518055, China (e-mail: swma@pku.edu.cn; cmjia@pku.edu.cn; zhzhao@pku.edu.cn; sswang@pku.edu.cn).

X. Zhang is with the Department of Computer Science, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: zhangxinf07@gmail.com).

S. Wang is with the Department of Computer Science, City University of Hong Kong, Hong Kong (e-mail: shiqwang@cityu.edu.hk).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2019.2910119

I. INTRODUCTION

IMAGE and video compression plays an important role in providing high quality image/video services under the limited capabilities of transmission networks and storage. The redundancies within images and videos are fundamentally important for image and video compression, including spatial redundancy, visual redundancy and statistical redundancy. Besides, the temporal redundancy existing in video sequences enables the video compression to achieve higher compression ratio compared with image compression.

For image compression, the early methods mainly realize compression by directly utilizing the entropy coding to reduce statistical redundancy within the image, such as Huffman coding [1], Golomb code [2] and arithmetic coding [3]. In later 1960s, transform coding was proposed for image compression by encoding the spatial frequencies, including Fourier transform [4] and Hadamard transform [5]. In 1974, Ahmed *et al.* proposed Discrete Cosine Transform (DCT) for image coding [6], which can compact image energy in the low frequency domain such that compression in the frequency domain becomes much more efficient.

Besides reducing statistical redundancy by entropy coding and transform techniques, the prediction and quantization techniques are further proposed to reduce spatial redundancy and visual redundancy in images. The most popular image compression standard, JPEG, is a successful image compression system by integrating its preceding coding techniques. It first divides image into blocks and then transforms blocks into the DCT domain. For each block, the differential pulse code modulation (DPCM) [7] is applied to its DC components, such that the prediction residuals of DC components between neighboring DCT blocks are compressed instead of compressing the DC value directly. To reduce the visual redundancy, a special quantization table is designed to well preserve low-frequency information and discard more high-frequency (noise-like) details as humans are less sensitive to the information loss in high frequency parts [8]. Another well-known still image compression standard, JPEG 2000 [9], applies the 2D wavelet transform instead of DCT to represent images in a compact form, and utilizes an efficient arithmetic coding method, EBCOT [10], to reduce the statistical redundancy existing in wavelet coefficients.

For video coding, temporal redundancy, which could be removed by inter-frame prediction, becomes the dominant one
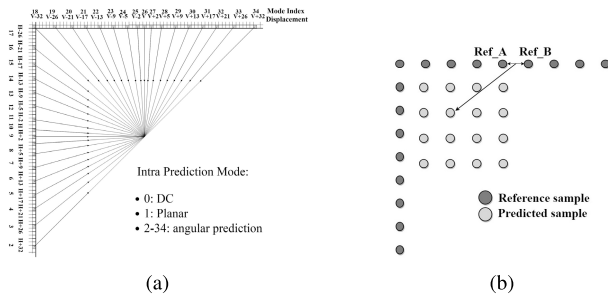
Fig. 1. Illustration of HEVC intra prediction. (a) different intra modes; (b) angular prediction instance.

due to the high correlation between successive frames captured in a very short time interval. To acquire inter-prediction efficiently, the block based motion prediction was proposed in 1970s [11]. In 1979, Netravali and Stuller proposed motion compensation transform framework [12], which is well known as the hybrid prediction/transform coder nowadays. Reader provided an introduction to the historical development of the first generation methods [13].

After several decades of development, the hybrid prediction/transform coding methods have achieved great success. Many coding standards have been developed and widely used in various applications, such as MPEG-1/2/4, H.261/2/3 and H.264/AVC [14], as well as AVS (Audio and Video coding Standard in China) [15] and HEVC [16]. Taking the latest video coding standard, HEVC, as an example, it utilized neighboring reconstructed pixels to predict the current coding block, with 33 angular intra prediction modes, the DC mode and the planar mode, as shown in Fig. 1. For inter-frame coding, HEVC improves the coding performance by further refining its predecessor, H.264/AVC, from multiple perspectives, e.g., increasing the diversity of the PU division, utilizing more interpolation filter taps for sub-sample motion compensation [17] and refining the side information coding including more most probable modes (MPMs) for intra mode coding [18], advanced motion vector prediction (AMVP) and merge mode for motion vector predictor coding [19]. Another new video coding tool in state-of-the-art video coding framework is loop filtering, and many loop filters [20]–[25] have been proposed since 2000. Herein, the deblock filtering [26], [27] and sample adaptive offset (SAO) [28] has been adopted into HEVC. However, the refinement strategies for traditional hybrid video coding framework based on image and video local correlations are more and more difficult for further coding efficiency improvement.

Recently, neural networks, especially the convolution neural networks (CNN), have achieved significant success in many fields including the image/video understanding, processing and compression etc. **A CNN is usually comprised of one or more convolutional layers. In particular, some tasks also append several fully connected layers after the convolution layers**. The parameters in these layers can be well trained based on massive image and video samples labelled for specific tasks in an end-to-end strategy. The trained CNN can be well applied to solve classification, recognition and prediction tasks on test data with highly efficient adaptability. The quality

of the prediction signals generated by CNN has surpassed that of the rule-based predictors. Moreover, the CNN can be interpreted as feature extractors to transform the image and video into feature space with compact representation, which is beneficial for image and video compression. Based on these excellent characteristics of CNN, it has also been recognized as a promising solution for compression task. Therefore, to well understand the existing development of CNN on image and video compression, this paper provides a detailed review on image and video compression using neural network.

Due to the vast scope of this review, we divide the main body of the paper into four parts for clearer presentation. In section II, we introduce the basic concept for neural network and image/video compression. Section III provides a detailed review on the development of neural network based image compression techniques. In section IV, we review the techniques of neural network based video compression. In section V, we revisit the neural network based optimization techniques for image and video compression. The further rationale in section III mainly follows the timeline of network development to introduce the neural network based image compression based on representative network architectures. In section IV, we mainly focus on the CNN based video coding techniques imbedded in the state-of-the-art hybrid video coding framework, HEVC, and also introduce some new video coding frameworks based CNN. Finally, section VI prospects the important challenges in deep learning based image/video compression and concludes the paper.

II. INTRODUCTION OF NEURAL NETWORK AND IMAGE/VIDEO COMPRESSION

In this section, we firstly revisit the basic concepts and development history of neural networks briefly. Subsequently, we introduce the frameworks and basic technique development for block based image coding and hybrid video coding framework.

A. Neural Network

With the interdisciplinary research of neuroscience and mathematics, the neural network (NN) was invented, which has shown strong abilities in the context of non-linear transform and classification. Intuitively, the network consists of multiple layers of simple processing units called neuron (perceptron), which interacts with each other via weighted connections. The neurons get activated through weighted connections from previously activated neurons. To achieve non-linearity, the activation functions are always applied for all the intermediate layers [29]. A simple neural network architecture is shown in Fig. 2, which consists of one input layer, one output layer and multiple hidden layers, each of which contains various number of neurons.

The learning procedure of simple perceptron has been proposed and analyzed in 1960s [30]. During the 1970s and 1980s, backpropagation procedure [31], [32] inspired by the chain rule for derivatives of the training objectives was proposed to solve the training problem of the multi-layer perceptron (MLP). Then, the multi-layer architectures
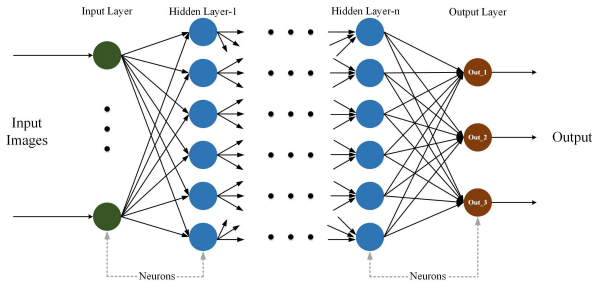
Fig. 2.   Illustration of the neural network architecture.

are mostly trained by stochastic gradient descent with back-propagation procedure although it is computationally intensive and suffers from bad local minima. However, the dense connections between the adjacent layers in neural networks make the amount of model parameters increase quadratically and prohibit the development of neural networks in computational efficiency. With the introduction of parameter-sharing for MLP 1990 [33], a more light-weighted version of neural network called convolutional neural network was proposed and applied in the documents recognition, which makes the large scale neural network training possible.

B. Image and Video Compression

Among the various coding frameworks, the core techniques in image and video compression are transform and prediction. JPEG [34] is the most popular image compression standard, which consists of the basic transform/prediction modules as shown in Fig. 3(a). In JPEG, the input image is partitioned into 8×8 non-overlapped blocks, each of which is transformed into the frequency domain using block-DCT (BDCT). For each transformed block, the DCT coefficients are then compressed into a binary stream via quantization and entropy coding. For video compression, most of popular video coding standards adopt the transform-prediction based hybrid video coding framework as shown in Fig. 3(b), e.g., MPEG-2, H.264/AVC and HEVC. Different from JPEG, HEVC utilizes more intra prediction modes from neighboring reconstructed blocks in spatial domain instead of DC prediction, as shown in Fig. 1. Besides intra prediction, more coding gains of video compression come from the high efficient inter prediction, which utilizes motion estimation to find the most similar blocks as prediction for the to-be-coded block. Moreover, HEVC adopts two loop filters, i.e., deblocking filter and SAO, to reduce the compression artifacts sequentially.

In the above block based image and video coding standards, the compression is usually block-dependent and must be performed block by block sequentially, which limits the compression parallelism using parallel computation platform, e.g. GPU. Moreover, the independent optimization strategy for each individual coding tool also limits the compression performance improvement compared with end-to-end optimization compression. In essence, there is another technological development trajectory based on the neural network techniques for image and video compression as summarized in Fig. 4. With the resurgence of neural network, the marriage of traditional image/video compression and CNN further advances

their progress. In the following sections, we will introduce the development of neural network based image/video compression and related representative techniques.

III. PROGRESS OF NEURAL NETWORK BASED IMAGE COMPRESSION

In this section, we introduce the image compression using machine learning methods especially from neural network perspective, which mainly originated from late 1980s [35]. This section is organized according to the historical development of neural network techniques, mainly including the Multi-layer Perceptron (MLP), Random Neural Network, Convolutional Neural Network (CNN) and Recurrent Neural Networks (RNN). In the final subsection, we will introduce the recent development of the image coding techniques using generative adversarial networks (GAN).

A. Multi-Layer Perceptron Based Image Coding

MLP [36] consists of an input layer of neurons (or nodes, units), several hidden layers of neurons, and a final layer of output neurons. The output h_i of each neuron i within the MLP is denoted as,

$$h_i = \sigma \left(\Sigma_{j=1}^{N} w_{ij} x_j + c_i \right), \tag{1}$$

where $\sigma(\cdot)$ is the activation function, c_i denotes the bias-term of linear transform and the w_{ij} indicates the adjustable parameter, *weight*, which represents the connection between layers. The theoretical analysis has shown that the MLP constructed with over one hidden layer can approximate any continuous computable function to an arbitrary precision [37]. This property provides the evidence for the scenarios such as dimension reduction and data compression. The initiative of using MLP for image compression is to design unitary transforms for the whole spatial data.

In 1988, Chua and Lin proposed an end-to-end image compression framework by leveraging high parallelism and the powerful compact representation of neural network [35], which may be useful as a model of the human brain-like coding functions. They formulated the traditional image compression steps, i.e., the unitary transform of spatial domain image data, the quantization of transform coefficients and binary coding of quantized coefficients, as an integrated optimization problem to minimize the following cost function,

$$e_{k,l} = \| \mathbf{X} - \hat{y}_{kl} \mathbf{u}_k^T \mathbf{u}_l \|^2 \tag{2}$$

$$\hat{y}_{kl} = (s_1 2^{-1} + s_2 2^{-2} + \cdots + s_{kl} 2^{-b_{kl}}) - m_{k,l}, \tag{3}$$

where $\hat{y}_{kl} \in [0, 1]$ is the reconstructed transform coefficients, $\mathbf{u}_k$ is the orthogonal transform kernel and $\{s_1, s_2, \cdots, s_{b_{kl}}\}$ are the binary codes representing the quantization level for $\hat{y}_{kl}$. Then, the authors utilized a decomposition/decision neural network to solve the optimization problem in Eqn. (2) to find the optimal binary code combination, which is the output of the compressed bitstream. In 1989, the fully connected neural network with 16 hidden units was trained to compress each 8×8 patch of an image using back propagation [39]. However, this strategy fixed the neural network parameters for specific
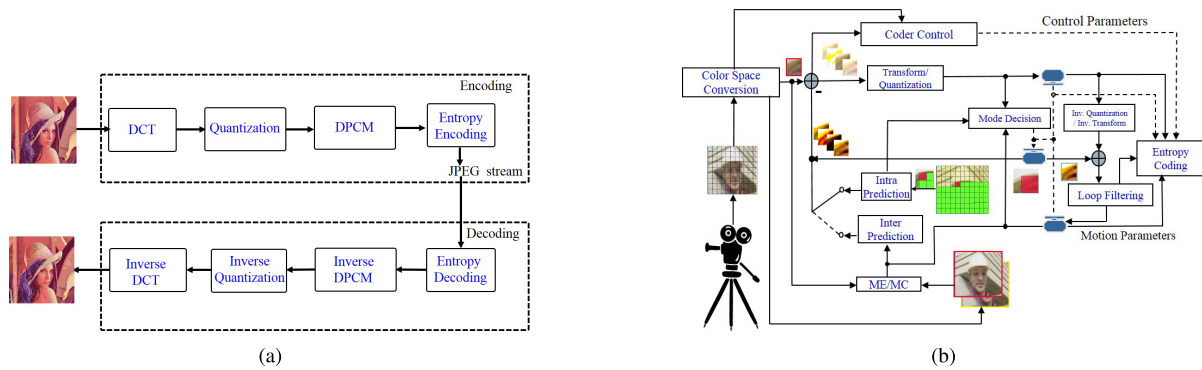
Fig. 3. Image and video compression framework (a) JPEG compression, (b) hybrid video compression.
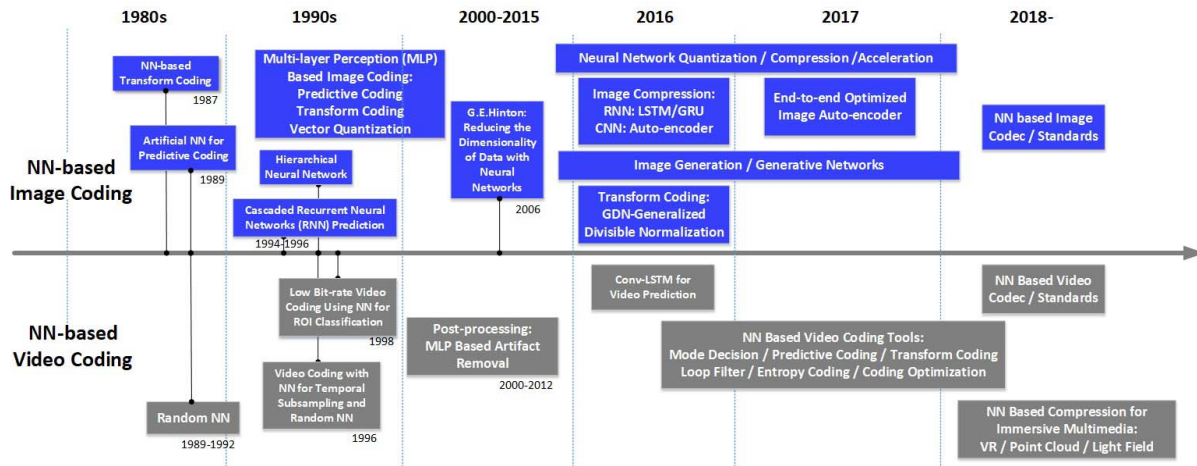
Fig. 4. The technical roadmap of neural network based compression algorithms.
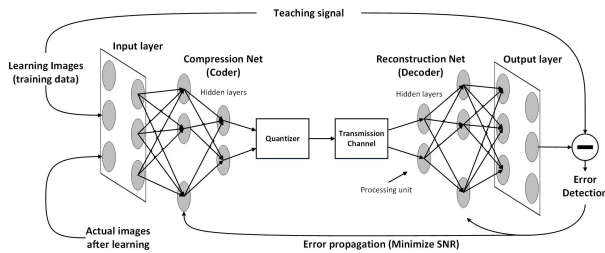
Fig. 5. The neural network based image codec [38].

number of binary codes, which is difficult to adapt to variable compression ratio in the optimal state.

Sonehara *et al.* proposed to train a dimension reduction neural network to compress the input image, and took the quantization and entropy coding as individual modules [38]. Fig. 5 shows the architecture of the dimension reduction neural network, where the auto-encoder bottleneck structure is deployed. In particular, the number of neurons in the bottleneck layer is smaller than the number of neurons in the input and output layers so as to reduce the dimension of data. To speed up the learning process, the input image is divided into blocks, which are fed to different sub-neural networks in parallel. The design using multiple sub-neural networks requires the input image to be strictly similar to the learned ones due to different sub-neural networks being responsible

for texture-specific structures. Therefore, the generalization of this model is limited, specifically, a loss of up to 10dB in SNR for unlearned images is reported. To obtain better performance and generalization capability, Sicuranza *et al.* trained a unique small neural network by feeding the image blocks sequentially, the loss of which is only about 1dB in SNR from unlearned images to learned images [40].

However, the adaptivity of above mentioned algorithms is determined by manually-setting different number of hidden neurons rather than bringing networks with more layers and complex connections, which may restrict the power of MLP in terms of compression performance [41]. To tackle this problem, MLP-based predictive image coding algorithm [42] was investigated by exploiting the spatial context information. Specifically, the spatial information to the left and above (points A, B and C, each small block corresponds to one pixel in Fig. 6(a)) was adopted to generate the non-linear predictor of the bottom-right pixel X in Fig. 6(a). There are three input nodes, 30 hidden nodes and one output node for the MLP predictor as shown in Fig. 6(b), and the MLP model is trained by utilizing the back propagation algorithm [32] to minimize the mean square errors between original and predicted signals. Based on their experiments, the MLP based non-linear predictor achieves an improvement on error entropy from 4.7 bits per pixel (bpp) to 3.9 bpp compared with linear predictor.
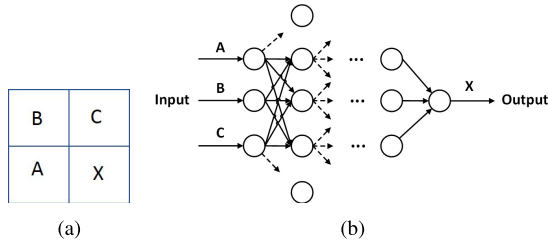
Fig. 6. Illustration of MLP-based predictive coding for image compression [42].

Fig. 7. The parameterized architecture of the CNN based end-to-end image compression proposed in [52].

To further improve the prediction accuracy, Manikopoulos utilized a high-order prediction model as in Eqn.(4) for a generalized auto-regressive (AR) model, which can well handle the sharply defined structures such as edges and contours in images [43].

$$x_n = \Sigma_i w_i x(n-i) + \Sigma_i \Sigma_j w_{ij} x(n-i) x(n-j)$$
$$+ \Sigma_i \Sigma_j \Sigma_k w_{ijk} x(n-i) x(n-j) x(n-k) + \ldots + \epsilon_n,$$
$$(4)$$

where ϵ_n is a sequence of zero-mean i.i.d. random variables. In 1996, the hierarchical neural network with its Nested Training Algorithm (NTA) was proposed for MLP based image compression [44], which considerably reduced the training time. The interested reader can refer to [45], [46] to known more MLP based image compression techniques, which improve the compression efficiency by designing different connection strucures.

B. Random Neural Network Based Image Coding

A new class of random neural network [47] was introduced in the 1989. Random neural network performs differently from the above mentioned MLP based methods in which signals are in spatial domain and optimized by the gradient back-propagation method. The signals in random neural network are transmitted in the form of spikes of unit amplitude. The communication between these neurons is modeled as a Poisson process where positive signals represent excitatory signals and negative signals represent inhibition. Some theoretical results were presented to analyze the behavior of random neural network in [47]. A "backpropagation" type training method is adopted to update the parameters, which requires the solution of n linear and n non-linear equations each time with a new input-output pair.

Some researchers considered the combination of the random neural network and image compression, and presented some meaningful results. Gelenbe and Sungur first applied the random neural network in the image compression task [48]. The architecture adopts a feedforward encoder/decoder random neural network with one intermediate layer. In particular, the first layer takes an image as the input, the last layer outputs a reconstructed image and the intermediate layer products compressed bits. Gelenbe and Sungur further extend the work in [48] by designing a adaptive still block-by-block random neural network compression/decompression [49]. There are multiple distinct neural compression networks $C_1,\ldots,C_L$ which are designed to achieve different compression levels. Each of
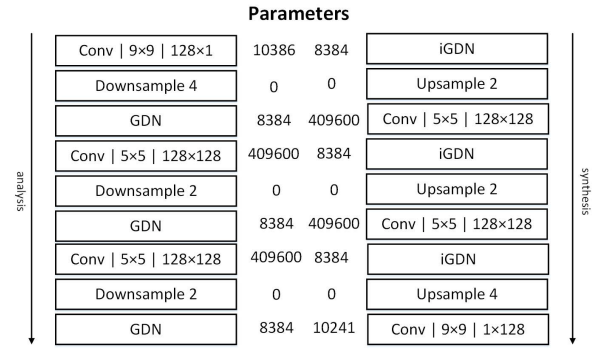
these networks compresses the block in parallel, and the choice of the networks is select according to the quality of decompressed results. Hai further improved the compression performance by integrating the random neural network into the wavelet domain of images [50].

C. Convolutional Neural Network Based Coding

Recently, CNN outperforms the traditional algorithms by a huge margin in high-level computer vision tasks such as the image classification, object detection [51]. Even for many low-level computer vision tasks, it also achieves very impressive performance, e.g., super-resolution and compression artifact reduction. CNN adopts the convolution operation to characterize the correlation between neighboring pixels, and the cascaded convolution operations well conform the hierarchical statistical properties of natural images. In addition, the local receptive fields and shared weights introduced by the convolution operations also decrease trainable parameters of CNN, which significantly reduce the risk of the over-fitting problem. Inspired by powerful representation of CNN for images, many works have been carried out to explore the feasibility of CNN-based lossy image compression.

However, it is difficult to straightforwardly incorporate the CNN model into end-to-end image compression. Generally speaking, CNN training depends on the back-propagation and stochastic gradient descent algorithm which demand the almost-everywhere differentiability of the loss function with respect to the trainable parameters such as the convolution weights and biases. Due to the quantization module in image compression, it produces zero gradients almost everywhere which stops the parameters updating in the CNN. In addition, the classical rate-distortion optimization is difficult to be applied to CNN based compression framework. This is because the end-to-end training for CNN needs a differentiable loss function, but the rate must be calculated based on the population distribution of the whole quantized bins, which is usually non-differentiable with respect to arguments in CNN.

Ballé *et al.* first introduced an end-to-end optimized CNN framework for image compression under the scalar quantization assumption in 2016 [52], [53]. The framework is illustrated in Fig. 7, which consists of two modules, *i.e.,* analysis and synthesis transforms for encoder and decoder. In analysis transform, there are three stages of convolution, subsampling,

and divisive normalization. Each stage starts with an affine convolution:

$$v_i^{(k)}(m, n) = \sum (h_{k,ij} * u_j^{(k)})(m, n) + c_{k,i}, \qquad (5)$$

where $u_j^{(k)}$ is the jth input channel of the kth stage at spatial location (m, n), $*$ denotes 2D convolution operation and $h_{k,ij}$ represents the convolution parameter. $c_{k,i}$ is the bias parameter of the convolution neural network. The output of convolution is downsampled:

$$w_i^{(k)}(m, n) = v_i^{(k)}(s_k m, s_k n), \qquad (6)$$

where s_k is the downsampling factor. Finally, the downsampled signals processed by a generalized divisive normalization (GDN) transform:

$$u_i^{(k+1)}(m, n) = \frac{w_i(k)(m, n)}{(\beta_{k,i} + \sum_j \gamma_{k,ij}(w_j^{(k)}(m, n))^2)^{\frac{1}{2}}}. \qquad (7)$$

where $\beta_{k,i}$ and $\gamma_{k,ij}$ are the bias and scale parameters for the normalization operation.

Since the synthesis transform is the inverse operation of the analysis transform, all the parameters across all three stages, $\{h, c, \beta, \gamma\}$ will be optimized according to rate-distortion objective function in an end-to-end style. To deal with the zero derivatives due to the quantization, Ballé et al. utilized an additive i.i.d uniform noise to simulate the quantizer in CNN training procedure, which enables the stochastic gradient descent approach to the optimization problem. This method outperforms JPEG2000 according to both PSNR and MS-SSIM metrics. In addition, Ballé and his colleagues extended such model using the scale hyper priors for entropy estimation [54], which achieved similar objective coding performance with HEVC. Minnen et al. continued to enhance the context model of entropy coding for end-to-end optimized image compression [55] and outperformed the HEVC intra coding. For future practical utility, both hardware-end support and the energy-efficiency analysis should be further explored since the autoregressive component is not easily parallelizable. The image compression performance is further improved by Zhou et al. by utilizing pyramidal feature fusion structure at the encoder and a CNN based post-processing filter at the decoder [56]. The other end-to-end image compression work joint with quantization and entropy coding can be referred in [57] and [58], and the CNN prediction based image compression can be can be referred in [59].

D. Recurrent Neural Network Based Coding

Unlike the CNN architecture mentioned above, RNN is a class of neural network with memory to store the recent behaviors. In particular, memory units in RNN have the connections to themselves, which transmit transformed information from the execution in the past. By taking advantage of these stored information, RNN changes the behavior of the current forward process to adapt to the context of current input. Hochreiter et al. proposed the Long Short-Term Memory (LSTM) [60] to overcome the insufficiency of the decayed error backflow. More advanced improvements such as Gated Recurrent Unit (GRU) [61] are presented to simplify the
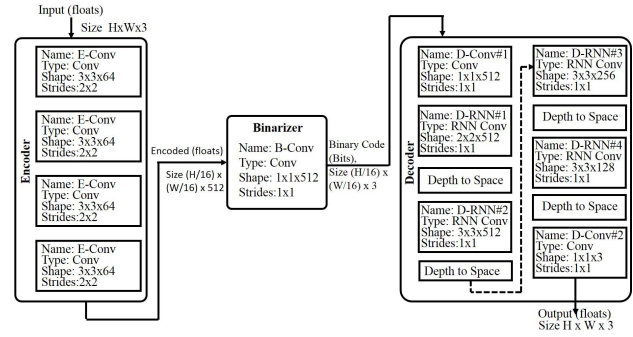

Fig. 8. A single iteration of our shared RNN architecture [63].

recurrent evolution processes, and meanwhile they maintain the performance of the recurrent network in relevant tasks [62]. In analogous to CNN, for image compression task, RNN still suffers from the difficulties to propagate the gradients of the rate estimation.

Toderici et al. firstly proposed a RNN-based image compression scheme [63] by utilizing a scaled-additive coding framework to restrict the number of coding bits instead of the approximation of rate estimation in CNN [52]. More specifically, the proposed method in [63] is an multi-iteration compression architecture supporting variational bitrate compression in progressive style. As shown in Fig. 8, there are three modules in a single iteration, i.e., an encoding network E, a binarizer B and a decoding network D, where D and E contain recurrent network components. The residual signals between the input image patch and the reconstructed one from decoding network D can be further compressed into the next iteration. To further improve the RNN-based image compression, Minnen et al. presented a spatially adaptive image compression framework [64]. In this framework, the input images is divided into tiles which is similar to the existing image codecs such as the JPEG and JPEG2000. For each tile, an initial prediction is generated by a fully-convolutional neural network from the spatial neighboring tiles which have been decoded in the left and above regions. However, based on the released results, the proposed method only outperforms JPEG while it is inferior to JPEG2000.

E. Generative Adversarial Network Based Coding

Generative Adversarial Network is one of most attractive improvements in the application of deep neural network. GAN optimizes two network models i.e., generator and discriminator, simultaneously. Discriminator takes advantage of deep neural network to distinguish whether the samples are generated form the generator. At the same time, the generator is trained to overcome the discriminator and produce samples which pass the inspection. Adversarial loss has the advantage to assist the generator to improve the subjective quality of images and also can be designed for different tasks. In the image compression task, some research works focused on the perceptual quality of the decoded images and utilized GAN to improve the coding performance.

One of the representative works is proposed by Rippel and Bourdev in 2017 [65], and it is an integrated and
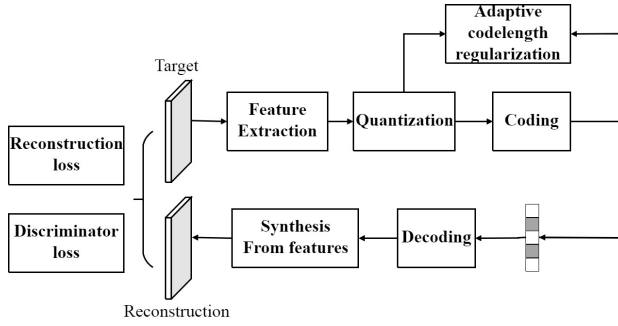
Fig. 9. The overall architecture of GAN based image compression [65].

well optimized GAN based image compression, which not only achieves amazing compression ratio improvement but also can run in real-time by leveraging the massive parallel computation cores of GPU. As shown in Fig. 9, the input image is compressed into very compact feature space by networks as its compressed form, and the generative network is utilized to reconstruct the decoded image from the features. The most obvious difference between the GAN based image compression and those of CNN or RNN based schemes is the introduction of the adversarial loss which enhances the subjective quality of reconstructed image significantly. The generative network and adversarial network are trained jointly to significantly enhance the performance of the generative model. The GAN based method in [65] achieves significant compression ratio improvement, e.g., producing compressed files 2.5 times smaller than JPEG and JPEG2000, 2 times smaller than WebP, and 1.7 times smaller than BPG on generic images across all quality levels. Herein, the quality is measured by MS-SSIM, while the method is still not efficient using PSNR metric. Inspired by the advances in GAN based view synthesis, the light field (LF) image compression could achieve significant coding gain with generating the missing views using the sampled context views in LF [66]. In particular, the contents generated by GAN are more consistent with the semantics of the original content than the specific textures. Especially, when enlarging the reconstructed images, we can see the content difference in specific textures.

In addition, Gregor *et al.* introduced a homogeneous deep generative convolutional model *DRAW* [67] to the image compression task. Different from previous works, Gregor *et al.* aimed to conceptual compression by generating the image semantic information as possible [68]. A GAN-based framework for extreme image compression, targeting bitrates below 0.1 bpp, is explored in detail, which allows for different degrees of content generation [69]. At present, the GAN-based compression is successful in narrow-domain images such as faces, and still needs more research on establishing models for general natural images.

IV. ADVANCEMENT OF VIDEO CODING WITH NEURAL NETWORKS

The study of deep learning based video coding by leveraging the state-of-the-art video coding standard HEVC has been an active area of research in recent years. Almost all the modules in HEVC have been explored and improved by incorporating various deep learning models. In this section, we will review the development of video coding works with deep learning models from the five main modules in HEVC, i.e., intra prediction, inter-prediction, quantization, entropy coding and loop filtering. Finally, we will introduce several novel video coding paradigms, which are different from hybrid video coding framework.

A. Intra Prediction Techniques Using Neural Networks

Although many neural network based image compression methods have been proposed and can be regarded as intra-coding strategy for video compression, their performances only surpass JPEG and JPEG2000 and are inferior to HEVC intra coding obviously. This also shows the superiority of the hybrid video coding framework. Therefore, many researchers focuses on video coding performance improvement by integrating the neural network techniques into hybrid video coding framework, especially into the state-of-the-art HEVC framework. Cui *et al.* proposed an intra-prediction convolutional neural network (IPCNN) to improve the intra prediction efficiency, which is the first work integrating CNN into HEVC intra prediction. In IPCNN, the current 8×8 block is firstly predicted according to HEVC intra prediction mechanism, and the best prediction version of current block generated by mode decision as well as its three nearest neighboring reconstructed 8×8 blocks as additional context, i.e., the left block, the upper block and the upper-left block, composes a 16×16 block, which is utilized as the input of IPCNN. The residual learning approach is adopted and the output of IPCNN is the residual block by subtracting the original blocks from the input ones. Then, the refined intra-prediction for the current 8×8 block can be derived by subtracting the output residual block from the input one. The designed IPCNN not only heritages the powerful prediction efficiency of CNN, but also takes advantage of the far-distance structure information in spatial neighboring 8×8 blocks instead of only utilizing one column plus one row reconstructed neighboring pixels as HEVC intra prediction. The additional context for prediction as well as the residue learning approach offers extra coding efficiency.

Instead of using CNN to improve the quality of best HEVC intra prediction, Li *et al.* proposed a new intra prediction mode using fully connected network (IPFCN) [70], which competes with the existing 35 HEVC intra prediction modes. Similar with IPCNN, IPFCN also utilizes neighboring multiple reference lines of reconstructed pixels as contextual input, but the prediction version of the current block from HEVC intra prediction is not utilized. Fig.10 shows the IPFCN structure, which is an end-to-end intra prediction mapping from reconstructed neighboring pixels to current block. Except for the output layer, each connected layer is followed by a nonlinear activation layer, where the parametric rectified linear unit (PReLU) is utilized. Each node of the output layer corresponds to a pixel. The corresponding coding performance as well as complexity is depicted in Table. I, where the abbreviation "L" means light (which means parameter reduction version of models w/o "L"), "D" means dual (which means train one particular IPFCN model for DC and Planar modes,
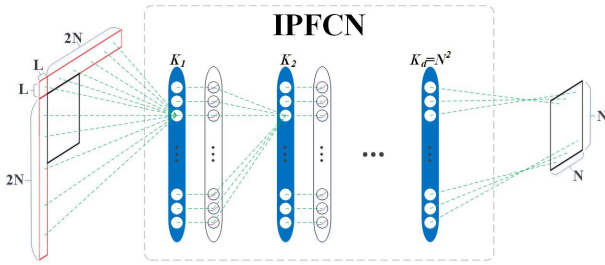
Fig. 10. The network structure of IPFCN [70].

Fig. 11. The framework of neural network based intra prediction using up-sampling [73].

TABLE I

THE CODING PERFORMANCE OF IPFCN [70] UNDER COMMON TEST CONDITION WITH FULL LENGTH SEQUENCE

Sequences	IPFCN vs. HM-16.9			
	IPFCN-S	IPFCN-D	IPFCN-S-L	IPFCN-D-L
Class A	-3.8 %	-4.4 %	-3.0%	-3.7%
Class B	-2.8 %	-3.2 %	-2.2%	-2.8%
Class C	-1.9 %	-2.1 %	-1.6%	-1.9%
Class D	-1.7 %	-1.8 %	-1.4%	-1.7%
Class E	-3.9 %	-4.5 %	-3.0%	-3.5%
Overall	-2.6 %	-3.0 %	-2.1%	-2.5%
Encode Time	4930%	13052%	285%	483%
Decode Time	26572%	28927%	923%	1141%

and another IPFCN model for the remaining angular modes), and "S" means single model (which means to train one model for all the intra prediction modes). The running time is tested on CPU platform. Compared with HEVC reference software HM-16.9, the proposed method achieved obvious bitrate saving, up to 3.0% BD-rate saving on average. In particular, the IPFCN performs better for ultra high resolution 4K videos in class A, achieving up to 4.4% BD-rate saving. However, the complexity is extremely high due to the fully connected neural nets and the float-point operations during the multiplication calculation, and there are up to more than 200 times increase for decoding as shown in Table I. The CNN based chroma intra prediction is proposed in [71] by utilizing both the reconstructed luma block and neighboring chrom blocks to improve intra chroma prediction efficiency. In [72], Pfaff *et al.* proposed a more high efficiency intra prediction network under JEM software, and the running time of its simplification version only increase by 74% and 38% for intra encoding and decoding process with about 2.26% BD-rate saving.

Instead of using neighboring reference samples to obtain block prediction, Li *et al.* explored CNN based down/up-sampling techniques as a new intra prediction mode for HEVC [73] and its extension for inter frame is proposed in [74]. Different from previous image-level down/upsampling techniques [75], [76], Li *et al.* designed the down/upsampling method in CTU-level and the framework is shown in Fig. 11. In down/up-sampling mode, each CTU is firstly down-sampled into low resolution version, which is then coded using HEVC intra coding method. The upsampling is applied for the reconstructed low resolution CTU to restore its original resolution. To remove the boundary artifacts, a second
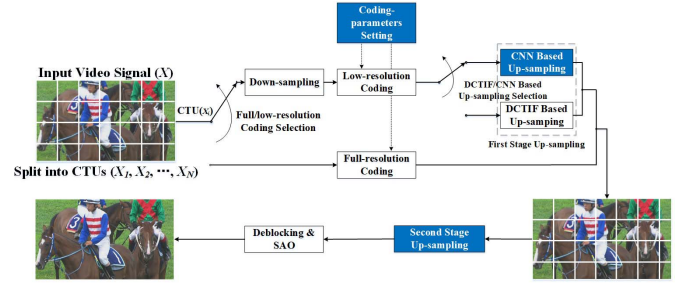
stage upsampling CNN network is applied when the whole frame has been reconstructed. Then, the second-stage upsampling CNN can access to all the surrounding blocks of down/upsampling CTUs. To ensure the coding efficiency, a flag is signaled into bitstream to indicate whether the down/upsampling is switched on. Herein, the flag is determined according to the rate distortion optimization at the encoder. Due to the high efficiency of CNN based upsampling techniques, this work achieves significant coding gain especially at low bitrate scenario, around 5.5% bitrate saving on average compared with HEVC. However, due to the limitations of the up-sampling algorithm, the bitrate saving for QPs (= 22, 27, 32, 37) utilized in common test condition of HEVC is only 0.7% for luma component.

In addition, the performance is also affected by the QPs used in compressed training video sequences, and the performance will degenerate when the test QPs deviate from those in the training stage. Based on the results using cross QP models, for the learned model from videos compressed at $QP = QP_0$, the performance degeneration when applying it to videos compressed at $QP_0 + 2$ is less than that of applying to videos compressed at $QP_0 - 2$. These results shows that the down/upsampling CNN prediction model is robust to the videos compressed by higher QPs. Beside intra prediction, Pfaff *et al.* utilized a fully connected neural network with one hidden layer and neighboring reconstructed samples to predict the intra mode probabilities [77], which can benefit the entropy coding module.

To alleviate the affects of compression noise on upsampling CNN, we proposed a dual-network based super-resolution strategy by bridging the low-resolution image and upsampling network using an enhancement network [78]. The enhancement network focuses on compressed noise reduction and feeds high quality input into the upsampling network. Compared with single upsampling network, the proposed method further improve coding performance at low bitrate scenario especially for ultra high resolution videos. In 2019, Li *et al.* designed a compact representation CNN model to further improve the super-resolution CNN based compression framework by constraining the information loss of the low resolution images [79]. Other CNN based intra coding techniques can be referred to [80], [81], wherein the CTU level CNN enhancement model for intra coding is introduced in [80] and RNN based intra prediction using neighboring reconstructed samples is introduced in [81].
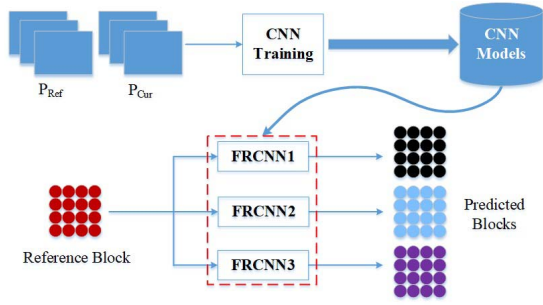
Fig. 12. Framework of the FRCNN in [84].

Fig. 13. Illustration for the proposed DVRF mode in [86].

B. Neural Network Based Inter Prediction

In hybrid video coding, the inter prediction is realized by motion estimation on previous coded frames against the current frame, and in HEVC the precision of motion estimation is up to quarter-pixel, the value of which is calculated via interpolation, e.g., discrete cosine transform based interpolation filter (DCTIF) [17]. Intuitively, the more similar of the inter predicted block and the current block are, the higher coding performance is achieved due to fewer prediction residuals left. Huo *et al.* proposed a straightforward method [82] to improve inter prediction efficiency by utilizing the existing variable-filter-size residue-learning CNN (VRCNN) [83], which is named CNN-based motion compensation refinement (CNNMCR). The CNNMCR jointly employs the motion compensated prediction and its neighboring reconstructed blocks as input of VRCNN, which is trained by minimizing the mean square errors between the input and its corresponding original signal. In fact, the improvement of CNNMCR for inter prediction is because the designed network can improve the inter prediction quality by reducing the compression noise and the boundary artifacts due to independent block processing.

Considering the importance of fractional-pixel motion compensation in inter prediction, Yan *et al.* proposed a Fractional-pixel Reference generation CNN (FRCNN) to predict the fractional pixels [84]. This work is different from the previous interpolation or super-resolution problems, which predict pixel values in high resolution image, while FRCNN is to generate the fractional-pixels from reference frame to approach the current coding frame. Therefore, the fractional-pixel generation is formulated as a regression problem with the loss function as,

$$f^* \equiv \arg\min_{f \in \triangle} \mathcal{L}(f(X), Y), \qquad (8)$$

where X is the motion compensation block by integer motion vector, Y is current coding block, and f is the regressor, which is implemented by CNN. Since the optimal position indicated by motion vectors may be in different fractional-pixel positions, e.g., three positions for half-pixel precision $(0,1/2)$, $(1/2,0)$ and $(1/2,1/2)$. Thus, an individual CNN is trained for each fractional-pixel positions. Fig.12 shows a training example for three half-pixel CNN models. In essence, the principle of FRCNN is the same with that of adaptive interpolation filters [85], the parameters of which are derived by minimizing the prediction errors at fractional-pixel positions and need to be transmitted to decoder side. The performance of FRCNN
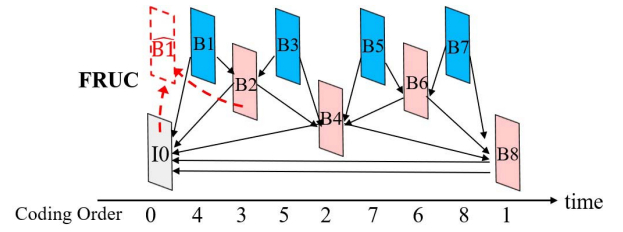
TABLE II
THE CODING PERFORMANCE OF CNN-BASED BI-PREDICTION AND DVRF

Sequences	BIP-CNN vs. HM-16.9		DVRF	
	RA	LDB	RA (HM16.9)	RA (JEM7.1)
Class A	-2.1 %	-1.7 %	-6.7%	-1.3%
Class B	-3.2 %	-1.9 %	-3.5%	-0.4%
Class C	-2.2 %	-0.9 %	-4.0%	-0.8%
Class D	-3.2 %	-1.0 %	-5.7%	-0.7%
Class E	/	-2.8 %	/	-0.8%
Overall	-2.7 %	-1.7 %	-4.6%	-0.7%
Encode Time	149%	185%	135%	124%
Decode Time	4259%	2853%	4376%	1025%

mainly thanks to the high prediction efficiency of CNN, and it achieves on average 3.9%, 2.7% and 1.3% bitrate saving compared to HM-16.7, under Low-Delay P (LDP), Low-Delay B (LDB) and Random-Access (RA) configurations, respectively. However, the performance improvement comes from up to 120 FRCNN models for different slice types and 4 common QPs, which are trained from the specific videos compressed by HEVC under 4 common QPs and various coding configurations. Due to the poor generalization of the CNN models, the performance of FRCNN model may degenerate when applying it to the videos compressed by different configurations and QPs from training data, which is a potential problem to be solved in future.

Instead of improving the prediction performance for fractional-pixels, we directly explore the inter prediction block generation using CNN based frame rate up conversion (FRUC) techniques. A CNN based FRUC method in CTU level is proposed to generate a virtual reference frame $F_{Virtual}$, which is utilized as a new reference frame and named as direct virtual reference frame (DVRF) [86], [87]. As shown in Fig. 13, current coding block can directly take the co-located block in $F_{Virtual}$ as inter prediction block without motion vectors. The state-of-the-art deep FRUC approach *Adaptive Separable Convolution* [88] is adopted and the two nearest bi-directional reference frames in the reference list are utilized as input for the network. This method achieves very promising compression performance, about 4.6% bitrate saving compared with HM-16.9 and 0.7% bitrate saving compared with JEM-7.1 [89] on average as shown in Table II. Herein, JEM (Joint Exploration Model) is the reference software based on HEVC reference model for the JVET group, which is an organization working on the exploration of next generation video coding standard established by ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG) in Oct. 2015. In addition, considering

the limitation of the traditional bidirectional prediction using simple average of two prediction hypothesises, we further improve its efficiency by utilizing a six-layer CNN with 13×13 receptive field size to infer the inter prediction block in a nonlinear fashion [90], [91], which achieves 2.7% bitrate saving compared with HM-16.19 on average under the RA configuration as shown in Table II. Although these methods obtained significant compression performance improvement, they also dramatically increase the run time for both encoding and decoding. The encoding and decoding time in Table II are all tested with GPU for the convolution calculations in the proposed methods. The computation efficiency is still a severe problem for CNN based video compression techniques in practical applications. The CNN based fractional interpolation methods can be referred to [92], [93].

C. Neural Network Based Quantization and Entropy Coding for Video Coding

In video coding, quantization and entropy coding are the lossy and lossless compression procedures respectively. For quantization part, the scalar quantization strategy has dominated hybrid video coding framework due to its low cost in computation and memory. However, this uniform scalar quantization does not conform to the characteristics of human visual system, and is not friendly to perceptual quality improvement. In [94], Alam *et al.* proposed a two-step quantization strategy using neural networks. In the first step, a CNN named VNet-2 is utilized to predict local visibility threshold C_T for HEVC distortions of individual video frames, and the VNet-2 consists of 894 trainable parameters in three layers, i.e., convolution, subsampling and full connection, each of which contains one feature map. In the second stage, the quantization steps for 64×64 CTU are derived by regression as,

$$\log(Q_{step}) = \alpha C_T^2 + \beta C_T + \gamma, \qquad (9)$$

where C_T is predicted visibility threshold in the first stage, and $\{\alpha, \beta, \gamma\}$ are the model parameters related with patch features. The model parameters are predicted from three separate committees of neural networks respectively, and each committee had a total of five two-layered feed-forward networks with 10 neurons. Based on the proposed adaptive quantization strategy, on average 11% bitrate saving can be obtained for luma channel against HEVC at the same perceptual quality measured by structural similarity (SSIM) [95].

After quantization, the syntax elements including coding modes and transform coefficients will be fed into entropy coding engine to further remove their statistical redundancy. HEVC adopts the CABAC as its entropy coding, which achieves very high coding efficiency mainly because there are many contexts designed to predict conditional probabilities accurately. Inspired by the prediction efficiency of CNN, Song *et al.* improved the CABAC performance on compressing the syntax elements of 35 intra prediction modes by leveraging CNN to directly predict the probability distribution of intra modes instead of the handcrafted context models [96]. The network architecture is based on LeNet-5 proposed by LeCun *et al.* [97], and the above-left, the above and the left

reconstructed blocks with the same size of current coding block are utilized as one category of inputs. The other category inputs are the most probable modes (MPMs), each of which are transformed into a 35-dim one-hot binary vector. The output is a 35-dim vector recording the predicted probability values of the 35 modes. Due to the high prediction accuracy, the CNN based method can improve the CABAC performance achieving about 9.0% bitrate saving for intra prediction mode coding when CU size is 8×8. The similar principle can be applied to other syntax elements, e.g., motion vector, coefficients and transform indices. Puri *et al.* applied the CNN to predict the optimal transform index probability distribution from the quantized coefficient blocks, and then utilized the probability to binarize the transform index using a variable length instead of a fixed length coding to improve the entropy coding performance [98]. At present, the works on entropy coding are still limited and remain to be investigated, especially there are few CNN based work on the dominant syntax, quantized transform coefficients in video coding, which may bring more coding gains.

D. Neural Network Based Loop Filtering

Loop filtering module is first introduced into video coding standard since H.263+ [99], and many different kinds of loop filters [21]–[23], [27], [28] are proposed after that. Especially, inspired by the success of CNN on image/video restoration filed, many of CNN based loop filters are designed to remove compression artifacts recently, which are much easier to implement the end-to-end training compared with other video coding modules. Zhang *et al.* [100] proposed a residual highway convolutional neural network (RHCNN) for loop filtering in HEVC. It is a deep network with 13 layers and the basic high way unit in each layer consists of two convolutional layers followed by the correponding activation function ReLUs and an identity skip connection. Since the compression noise levels are distinct for videos compressed with different QPs and frame types including I/B/P frames, the CNN models should be trained for different QP and frame type combinations, which lead to 156 CNN models for video coding application. To reduce memory cost for CNN based loop filters, Zhang *et al.* merged the QPs into several bands, and trained the optimal RHCNNs for each band. Compared with HM-12.0, the RHCNN achieves about 5.7%, 5.68% and 4.35% bitrate saving for I/P/B frames in low bit-rate circumstances respectively with 2~3 times of encoding time increase even using GPU and 20~30 times of encoding time increase using CPU.

By leveraging the coherence of the spatial and temporal adaptations, we improved the performance of CNN based loop filter, and designed the spatial-temporal residue network (STResNet) based loop filter [101]. The loss function of STResNet is formulated as,

$$L(\Theta) = \frac{1}{N} \sum_{i=1}^{N} \| F(x_{i-1}, x_i | \Theta) - y_i \|^2, \qquad (10)$$

where (x_{i-1}, x_i, y_i) are the training samples. x_{i-1} and x_i represent the $(i-1)^{th}$ and i^{th} reconstructed frames and y_i

TABLE III
THE CODING PERFORMANCE OF MUTI-MODEL CNN LOOP FILTER [102]

Sequences	Anchor HM-16.9			Anchor HM-16.9 + ALF		
	AI	LDB	RA	AI	LDB	RA
Class A	-4.7%	-6.7%	-6.6%	-2.7%	-3.2%	-3.1%
Class B	-3.5%	-5.7%	-6.5%	-1.6%	-2.5%	-2.7%
Class C	-3.4%	-5.0%	-4.5%	-3.4%	-4.0%	-3.7%
Class D	-3.2%	-3.8%	-3.3%	-3.2%	-3.4%	-3.4%
Class E	-5.8%	-8.6%	-9.0%	-4.3%	-5.8%	-5.3%
Overall	-4.1%	-6.0%	-6.0%	-2.9%	-3.7%	-3.6%
Encode Time	114%			108%		
Decode Time	15010%			12800%		

corresponds to the uncompressed i^{th} frame. $F(x_{i-1}, x_i|\Theta)$ represents the STResNet model, where Θ is the set of network parameters. Moreover, we further improved the filtering performance by introducing content-aware CNN based loop filter in [102]. For a reconstructed frame, multiple CNN models are trained according to their filtering performance iteratively as that in [103], and a corresponding discriminative network is also trained which is utilized to help select optimal filter in test stage to remove coding overheads. Compared with HEVC with/whitout ALF under HEVC common test condition (CTC), the proposed multi-model CNN filters achieve significant performance improvement as illustrated in Table III at the cost of explosive encoding and decoding run time increase even using GeForce GTX TITAN X GPU.

Although the CNN based loop filters have achieved substantial coding gains on the top of HEVC, these methods need to store multiple CNN models for different QPs, which increase the memory burdens for video codec. In [104], they provided an efficient solution for CNN based loop filters with memory efficiency. They combined QPs as an input fed into the CNN training stage by simply padding the scalar QPs into a matrix with the same size of input frames or patches. To some extent, this method alleviates the performance fluctuates of CNN based loop filters due to QP missing in training stage. Based on our experience, although the CNN based loop filters learned from combined QPs is a little inferior to QP-dependent CNN models, the performance loss is usually marginal. The residual prediction based CNN model of in-loop filter is proposed in [105] and the multi-scale CNN model for in-loop filter is designed in [106].

Regarding the complexity of DL and none-DL based loop filtering methods under HEVC framework, the encoding time of [102] is 114% and 108% when the ALF is turned off/on respectively. However, the deocding time is drastically increased into 15010% and 12800% respectively. While the corresponding encoding and decoding complexity for ALF itself is 104% and 123%. Hence, there still exists large quantity of space and potential in optimization for DL based loop filtering algorithms in future studies, such as pruning and quantization for the float-point weights and biases in neural networks.

Besides the in-loop filters, there are also some post-filtering algorithms proposed to improve the quality of decoded video and images by reducing the compression artifacts. Dong *et al.* proposed an end-to-end CNN [107] to remove the compression artifacts, which is learned in the supervised manner. The CNN architecture is work is derived from super-resolution network SRCNN [108] by embedding one or more "feature enhancement" layers after the first layer of SRCNN to clean the noisy features. Li *et al.* proposed a universal model to deal with compressed image at different compression ratios [109] by utilizing a very deep CNN model. Yang *et al.* proposed a multi-frame quality enhancement neural network for compressed video by utilizing the neighboring high quality frames to enhance the low quality frames. Herein, a support vector machine based detector is utilized to locate peak quality frames in compressed video [110]. CNN based quality enhancement also achieves convincing performance in the field of multiview plus depth video coding. Zhu *et al.* designed CNN models for the post-processing of synthesized views to promote the 3D video coding performances [111]. More works utilized more complicated structure to improve the compressed images [112], [113].

E. New Video Coding Frameworks Based on Neural Network

Although the elaborately designed hybrid video coding framework has achieved significant success on predominant compression performance, it becomes more and more difficult to be further improved. Moreover, it also becomes computation intensive and inhospitality to parallel computation as well as hardware manufacturer. Similar with neural network based image coding frameworks, some novel video coding frameworks are also investigated by assembling different neural network models. Chen *et al.* proposed a combination of several CNN networks called *DeepCoder* which achieved similar perceptual quality with low-profiled x264 encoder [114]. In *DeepCoder*, the intra prediction is implemented via a neural network to generate a feature map, denoted as *fMap*, and the inter prediction is obtained from motion estimation on previous frames. The *fMap* is further quantized and encoded into stream. The intra- and inter-prediction residuals are transformed into a more compact domain using neural networks, the process of which is similar with that of *fMap* generation in intra prediction but with different neural network parameters. Both the *fMaps* from intra prediction and residuals are quantized and coded using Huffman entropy coding. Although there are not as many coding tools as H.264/AVC, the *DeepCoder* shows comparable compression performance compared with H.264/AVC, which shows a new solution for video coding.

Chen *et al.* proposed a fully learning-based video coding framework by introducing the concept of VoxelCNN via exploring spatial-temporal coherence to effectively perform predictive coding inside learning network [115]. Specifically, the proposed video coding framework can be divided into three modules, i.e., predictive coding, iterative analysis/synthesis and binarization. The VoxelCNN is designed to predict blocks in the video sequences conditioned on previously coded frames as well as the neighboring reconstructed blocks of current block. Then the compact discrete representation of the difference between predicted and original signals can be analyzed and synthesized in iterative manner using RNN model of Toderici *et al.* [63], which is composed

of several LSTM-based auto-encoders with connections between adjacent stages. Finally, the bitstream is subsequently obtained after binarization and entropy coding. Although lack of entropy coding in their present work, the scheme still shows comparable performance with H.264/AVC, showing its potential in future video coding.

Inspired by the prediction for future frames of generative models [116], Srivastava *et al.* proposed to utilize the Long Short Term Memory (LSTM) Encoder-Decoder framework to learn video representations in [117], which can be utilized to predict future video frames. There are mainly two models, LSTM Autoencoder Model and LSTM Future Predictor Model, which consist of two recurrent neural networks. Different from Ranzato's work [116] predicting one future frame, this model can predict a long future sequence into the future. Based on the experiments, with 16 input natural video frames, the model can reconstruct these 16 frames and predict the future 13 frames.

V. Optimization Techniques for Image and Video Compression

The state-of-the-art video coding standard, HEVC, achieves the optimal compression performance by exhaustively traverse all the possible coding modes and partitions to determine the optimal coding parameters according to rate-distortion costs. The computational costs can be extremely reduced by predicting the optimal coding parameters to skip unnecessary RD calculations. The fast mode-decision algorithms are proposed for coding unit (CU) and prediction unit (PU) respectively on basis of neural networks, which are not only parallel-friendly but also easy for VLSI design [118], [119]. More specifically, the fast algorithm first carries out a coarse analysis based on the local gradients to classify the blocks into homogeneous and edge categories. This strategy not only can reduce the burden of CNN but also can make CNN avoid ill-conditions due to homogeneous blocks. Then, the CNN is designed for edge blocks to decrease no less than two CU partition modes in each CTU for full rate-distortion optimization process. The designed network contains one convolution layer with one max pooling layer followed by three full connected layers and takes the QP values into network at the last fully connected layer. Each square CU is used as network input while the output is the binary decision of quad-split or no-split for current CU. As such, the recursive mode traverse and selection process is eliminated. On average, their method achieves 61.1% intra coding time saving, whereas the BD rate loss is only 2.67% compared with HM-12.0. Xu *et al.* predicted the entire CTU partition structure by using both CNN and LSTM to determine whether the mode decision should be early terminated [120].

VI. Conclusions and Outlook

Image and video compression aims to seek more compact representation for visual signals while keeping high quality, and become more and more important in big visual data era. In this paper, the neural network based image and video compression techniques have been reviewed, especially for the recent deep learning based image and video compression techniques. With the survey presented earlier in this paper, it is apparent that the state-of-the-art neural network based end-to-end image compression is still in its infancy which only outperforms the JPEG2000 and struggles against HEVC. The marriage of neural network and traditional hybrid video coding framework obtained significant performance improvement compared with the latest video coding standard, HEVC. This demonstrates the advantages of both neural networks and hybrid video coding framework.

Based on the review, we think that the advantages of neural network in image and video compression are three folds. First, the excellent content adaptivity of neural network is superior to signal processing based model because the network parameters are derived based on lots of practical data while the models in the state-of-the-art coding standards are handcrafted based on image and video prior knowledge. Second, the larger receptive field is widely utilized in neural network models which not only utilizes the neighboring information but also can improve coding efficiency by leveraging samples from far distance, but the traditional coding tools only utilized the neighboring samples and are difficult to utilize far distant samples. Third, the neural network can well represent both texture and feature, which makes the joint compression optimization for both human view and machine vision analysis. However, the existing coding standards only pursue high compression performance toward human view task.

We envision that deep learning based image/video compression will play more important roles in representing and delivering images and videos with better quality and fewer bitrates, and the following confronted issues are required to be further investigated:

- **Semantic-fidelity oriented image and video compression.** Along with the fast development of computer vision techniques and explosively increasing of images and videos, the visual signal receivers are not only human visual system, but also the computer vision algorithms. Meanwhile, the neural network especially deep learning techniques are more appropriate for sematic information representation based on its great success in image and video understanding tasks. Therefore, the sematic-fidelity will become critical for further applications as well as traditional visual-fidelity requirement.

- **Rate-distortion (RD) optimization guided neural network training and adaptive switching for compression task.** The rate-distortion theory is the key of the success for traditional image and video compression, but it has not been well explored in current neural network based compression tasks. A single network to deal with all the images and videos with diverse structures is inefficient obviously. Therefore, the multi-network adaptively training and switching according to RD is a possible solution.

- **Memory and computation efficient design for practical image and video codec.** The biggest obstacle in hindering the deployment of deep learning based image and video compression is the burdens in computation and memory. To achieve high performance, larger neural

networks with more layers and nodes are usually considered, but the various efficiency of network parameters are not well explored. For image and video compression problem, at present, there is no related research work by jointly considering both the compression performance and the efficiency in computation and memory for neural networks, which is important for practical applications.

For the semantic-friendlily oriented image and video compression, we have attempted to design innovative visual signal representation framework to elegantly support both human vision viewing and machine vision analysis. In view of the lightweight and the importance of features for visual semantic descriptors, e.g., CNN features, we proposed the hierarchical visual signal representation in [121] by jointly compressing the feature descriptors and visual content. More specially, for each video frame, feature descriptors are first extracted and compressed, and then the decoded features are utilized to assist visual content compression by handling large-scale global motion. This strategy not only improves the visual content compression efficiency but also ensures the visual analysis performance due to the feature extraction from original video without the influence of compression artifacts. In [122], we further investigated the novel visual signal representation structure with deep learning based end-to-end image compression framework, which can directly conduct more image understanding tasks from the compression domain. The rationale behind this approach lies in that the neural network architectures commonly used for learned compression (in particular the encoders) are similar to the ones commonly used for inference, and learned image encoders are hence, in principle, capable of extracting features relevant for inference tasks. As such, this approach could be extended in the future to simultaneously train and learn for the end-to-end image compression and understanding.

In CNN based image and video compression, the CNN model compression is also a multi-variable optimization problem, which should be optimized jointly considering computational cost, CNN performance and rates utilized for CNN transmission (if needed). The previous work [123] proposed a complexity-distortion optimization formulation under power constraints for video coding problem, which can be further extended to CNN model compression optimization jointly with computational costs and video compression performance.

Based on the discussion of this paper, neural network has also shown promising results on future image and video compression tasks. Although there are still many problems in computational complexity and memory consumption, their high efficiency in prediction and compact representation for image and video signals has made neural network obtain substantial coding gain on top of the state-of-the-art video coding frameworks. Their intrinsic parallel-friendly attribute also makes them suitable for the largely deployed parallel computation architectures, e.g., GPU and TPU. Moreover, the network based end-to-end optimization approaches are more flexible than hand-crafted methods, and they can be rapidly optimized or tuned, which also makes the network with enormous potential in further image and video compression problem as well as other artificial intelligence problems.

REFERENCES

[1] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proc. Inst. Radio Eng.*, vol. 40, no. 9, pp. 1098–1101, Sep. 1952.

[2] S. Golomb, "Run-length encodings (corresp.)," *IEEE Trans. Inf. Theory*, vol. 12, no. 3, pp. 399–401, Jul. 1966.

[3] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, no. 6, pp. 520–540, Jun. 1987.

[4] H. Andrews and W. Pratt, "Fourier transform coding of images," in *Proc. Hawaii Int. Conf. Syst. Sci.*, 1968, pp. 677–679.

[5] W. K. Pratt, J. Kane, and H. C. Andrews, "Hadamard transform image coding," *Proc. IEEE*, vol. 57, no. 1, pp. 58–68, Jan. 1969.

[6] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. 100, no. 1, pp. 90–93, Jan. 1974.

[7] C. W. Harrison, "Experiments with linear prediction in television," *Bell System Tech. J.*, vol. 31, no. 4, pp. 764–783, Jul. 1952.

[8] G. K. Wallace, "Overview of the JPEG (ISO/CCITT) still image compression standard," *Image Process. Algorithms Techn.*, vol. 1244, pp. 220–234, Jun. 1990.

[9] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. Consum. Electron.*, vol. 46, no. 4, pp. 1103–1127, Nov. 2000.

[10] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Process.*, vol. 9, no. 7, pp. 1158–1170, Jul. 2000.

[11] Y. Taki, M. Hatori, and S. Tanaka, "Interframe coding that follows the motion," in *Proc. Inst. Electron. Commun. Eng. Jpn. Annu. Conv.(IECEJ)*, Aug. 1974, p. 1263.

[12] A. N. Netravali and J. A. Stuller, "Motion-compensated transform coding," *Bell Syst. Tech. J.*, vol. 58, no. 7, pp. 1703–1718, Sep. 1979.

[13] C. Reader, *History of Video Compression (Draft)*, document JVT-D068, 2002.

[14] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[15] (2018). *AVS Working Group Website*. [Online]. Available: http://www.avs.org.cn

[16] G. J. Sullivan, J.-R. Ohm, W.-J. 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.

[17] H. Lv, R. Wang, X. Xie, H. Jia, and W. Gao, "A comparison of fractional-pel interpolation filters in HEVC and H. 264/AVC," in *Proc. Vis. Commun. Image Process. (VCIP)*, Nov. 2012, pp. 1–6.

[18] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1792–1801, Dec. 2012.

[19] J.-L. Lin, Y.-W. Chen, Y.-W. Huang, and S.-M. Lei, "Motion vector coding in the HEVC standard," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 957–968, Dec. 2013.

[20] M. Naccari and F. Pereira, "Adaptive bilateral filter for improved in-loop filtering in the emerging high efficiency video coding standard," in *Proc. IEEE Picture Coding Symp. (PCS)*, May 2012, pp. 397–400.

[21] X. Zhang *et al.*, "Low-rank-based nonlocal adaptive loop filter for high-efficiency video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 10, pp. 2177–2188, Oct. 2017.

[22] S. Ma, X. Zhang, J. Zhang, C. Jia, S. Wang, and W. Gao, "Nonlocal in-loop filter: The way toward next-generation video coding?" *IEEE MultiMedia*, vol. 23, no. 2, pp. 16–26, Apr./Jun. 2016.

[23] C.-Y. Tsai *et al.*, "Adaptive loop filtering for video coding," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 934–945, Dec. 2013.

[24] X. Zhang, R. Xiong, S. Ma, and W. Gao, "Adaptive loop filter with temporal prediction," in *Proc. IEEE Picture Coding Symp. (PCS)*, May 2012, pp. 437–440.

[25] X. Zhang, S. Wang, Y. Zhang, W. Lin, S. Ma, and W. Gao, "High-efficiency image coding via near-optimal filtering," *IEEE Signal Process. Lett.*, vol. 24, no. 9, pp. 1403–1407, Sep. 2017.

[26] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, "Adaptive deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 614–619, Jul. 2003.

[27] A. Norkin *et al.*, "HEVC deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1746–1754, Dec. 2012.

[28] C.-M. Fu *et al.*, "Sample adaptive offset in the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1755–1764, Dec. 2012.

[29] G. E. Hinton, "Learning translation invariant recognition in a massively parallel networks," in *Proc. Int. Conf. Parallel Archit. Lang. Eur.*, 1987, pp. 1–13.

[30] F. Rosenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Washington, DC, USA: Spartan Books, 1962.

[31] P. Werbos, "New tools for prediction and analysis in the behavioral sciences," Ph. D. dissertation, Dept. Statist., Harvard Univ., Cambridge, MA, USA, 1974.

[32] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, p. 533, 1986.

[33] Y. Le Cun *et al.*, "Handwritten zip code recognition with multilayer networks," in *Proc. 10th Int. Conf. Pattern Recognit.*, Jun. 1990, pp. 35–40.

[34] G. K. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, no. 4, pp. 30–44, 1991.

[35] L. O. Chua and T. Lin, "A neural network approach to transform image coding," *Int. J. Circuit Theory Appl.*, vol. 16, no. 3, pp. 317–324, 1988.

[36] M. W. Gardner and S. R. Dorling, "Artificial neural networks (the multilayer perceptron)-a review of applications in the atmospheric sciences," *Atmos. Environ.*, vol. 32, nos. 14–15, pp. 2627–2636, Aug. 1998.

[37] R. J. Schalkoff, *Artificial Neural Networks*, vol. 1. New York, NY, USA: McGraw-Hill, 1997.

[38] N. Sonehara, M. Kawato, S. Miyake, and K. Nakane, "Image data compression using a neural network model," in *Proc. IJCNN*, vol. 2, 1989, pp. 35–41.

[39] P. Munro and D. Zipser, "Image compression by back propagation: an example of extensional programming," *Models cognition: rev. Cognit. Sci.*, vol. 1, no. 208, p. 1, 1989.

[40] G. L. Sicuranza, G. Romponi, and S. Marsi, "Artificial neural network for image compression," *Electron. Lett.*, vol. 26, no. 7, pp. 477–479, Mar. 1990.

[41] R. D. Dony and S. Haykin, "Neural network approaches to image compression," *Proc. IEEE*, vol. 83, no. 2, pp. 288–303, Feb. 1995.

[42] S. Dianat, N. Nasrabadi, and S. Venkataraman, "A non-linear predictor for differential pulse-code encoder (DPCM) using artificial neural networks," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Apr. 1991, pp. 2793–2796.

[43] C. N. Manikopoulos, "Neural network approach to DPCM system design for image coding," *IEE Proc. I Commun., Speech Vis.*, vol. 139, no. 5, pp. 501–507, Oct. 1992.

[44] A. Namphol, S. H. Chin, and M. Arozullah, "Image compression with a hierarchical neural network," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, no. 1, pp. 326–338, Jan. 1996.

[45] J. G. Daugman, "Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 7, pp. 1169–1179, Jun. 1988.

[46] H. M. Abbas and M. M. Fahmy, "Neural model for Karhunen-Loeve transform with application to adaptive image compression," *IEE Proc. I, Commun., Speech Vis.*, vol. 140, no. 2, pp. 135–143, Apr. 1993.

[47] E. Gelenbe, "Random neural networks with negative and positive signals and product form solution," *Neural Comput.*, vol. 1, no. 4, pp. 502–510, Dec. 1989.

[48] E. Gelenbe and M. Sungur, "Random network learning and image compression," in *Proc. IEEE Int. Conf. Neural Netw. (ICNN)*, vol. 6, Jun. 1994, pp. 3996–3999.

[49] C. Cramer, E. Gelenbe, and I. Bakircioglu, "Video compression with random neural networks," in *Proc. Int. Workshop Neural Netw. Identificat., Control, Robot. Signal/Image Process.*, Aug. 1996, pp. 476–484.

[50] F. Hai, K. F. Hussain, E. Gelenbe, and R. K. Guha, "Video compression with wavelets and random neural network approximations," *Proc. SPIE*, vol. 4305, pp. 57–65, Aug. 2001.

[51] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[52] J. Ballé, V. Laparra, and E. P. Simoncelli. (2016). "End-to-end optimized image compression." [Online]. Available: https://arxiv.org/abs/1611.01704

[53] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimization of nonlinear transform codes for perceptual quality," in *Proc. Picture Coding Symp. (PCS)*, Dec. 2016, pp. 1–5.

[54] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *Proc. Int. Conf. Learn. Represent.*, May 2018, pp. 1–10.

[55] D. Minnen, J. Ballé, and G. D. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 10771–10780.

[56] L. Zhou, C. Cai, Y. Gao, S. Su, and J. Wu, "Variational autoencoder for low bit-rate image compression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Aug. 2018, pp. 2617–2620.

[57] E. Agustsson *et al.*, "Soft-to-hard vector quantization for end-to-end learning compressible representations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1141–1151.

[58] L. Theis, W. Shi, A. Cunningham, and F. Huszär. (2017). "Lossy image compression with compressive autoencoders." [Online]. Available: https://arxiv.org/abs/1703.00395

[59] E. Ahanonu, M. Marcellin, and A. Bilgin, "Lossless Image Compression Using Reversible Integer Wavelet Transforms and Convolutional Neural Networks," in *Proc. IEEE Data Compress. Conf.*, Aug. 2018, pp. 1–10.

[60] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[61] K. Cho, B. van Merriénboer, D. Bahdanau, and Y. Bengio. (2015). "On the properties of neural machine translation: Encoder-decoder approaches." [Online]. Available: https://arxiv.org/abs/1409.1259

[62] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. (2014). "Empirical evaluation of gated recurrent neural networks on sequence modeling." [Online]. Available: https://arxiv.org/abs/1412.3555

[63] G. Toderici *et al.*, "Full resolution image compression with recurrent neural networks," in *Proc. CVPR*, Apr. 2017, pp. 5435–5443.

[64] D. Minnen *et al.* (2018). "Spatially adaptive image compression using a tiled deep network." [Online]. Available: https://arxiv.org/abs/1802.02629

[65] O. Rippel and L. Bourdev. (2017). "Real-time adaptive image compression." [Online]. Available: https://arxiv.org/abs/1705.05823

[66] C. Jia, X. Zhang, S. Wang, S. Wang, S. Pu, and S. Ma, "Light field image compression using generative adversarial network based view synthesis," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, to be published.

[67] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra. (2015). "Draw: A recurrent neural network for image generation." [Online]. Available: https://arxiv.org/abs/1502.04623

[68] K. Gregor, F. Besse, D. J. Rezende, I. Danihelka, and D. Wierstra, "Towards conceptual compression," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3549–3557.

[69] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, and L. Van Gool, "Extreme learned image compression with GANs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Sep. 2018, pp. 2587–2590.

[70] 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. 3236–3246, Jul. 2018.

[71] Y. Li *et al.*, "A hybrid neural network for chroma intra prediction," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 1797–1801.

[72] J. Pfaff *et al.*, *Intra Prediction Modes Based on Neural Networks*, document JVET-J0037, Apr. 2018, pp. 1–14.

[73] Y. Li *et al.*, "Convolutional neural network-based block up-sampling for intra frame coding," *IEEE Trans. Circuits Syst. for Technol.*, vol. 28, no. 9, pp. 2316–2330, Sep. 2017.

[74] J. Lin, D. Liu, H. Yang, H. Li, and F. Wu, "Convolutional neural network-based block Up-sampling for HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[75] R. Molina, A. K. Katsaggelos, L. D. Alvarez, and J. Mateos, "Toward a new video compression scheme using super-resolution," *Proc. SPIE*, vol. 6077, p. 607706, Jan. 2006.

[76] M. Shen, P. Xue, and C. Wang, "Down-sampling based video coding using super-resolution technique," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 6, pp. 755–765, Jun. 2011.

[77] J. Pfaff *et al.*, "Neural network based intra prediction for video coding," *Proc. SPIE*, vol. 10752, Aug. 2018, Art. no. 1075213.

[78] L. Feng, X. Zhang, X. Zhang, S. Wang, R. Wang, and S. Ma, "A dual-network based super-resolution for compressed high definition video," in *Pacific-Rim Conf. Multimedia*. New York, NY, USA: Springer, 2018, pp. 600–610.

[79] Y. Li, D. Liu, H. Li, L. Li, Z. Li, and F. Wu, "Learning a convolutional neural network for image compact-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1092–1107, 2019.

[80] Z.-T. Zhang, C.-H. Yeh, L.-W. Kang, and M.-H. Lin, "Efficient CTU-based intra frame coding for HEVC based on deep learning," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, Dec. 2017, pp. 661–664.

[81] Y. Hu, W. Yang, S. Xia, W.-H. Cheng, and J. Liu, "Enhanced intra prediction with recurrent neural network in video coding," in *IEEE Data Compress. Conf. (DCC)*, 2018, p. 413.

[82] S. Huo, D. Liu, F. Wu, and H. Li, "Convolutional neural network-based motion compensation refinement for video coding," in *Proc. Int. Symp. Circuits Syst. (ISCAS)*, May 2018, pp. 1–4.

[83] Y. Dai, D. Liu, and F. Wu, "A convolutional neural network approach for post-processing in HEVC intra coding," in *Proc. Int. Conf. Multimedia Model.*, 2017, pp. 28–39.

[84] N. Yan, D. Liu, H. Li, B. Li, L. Li, and F. Wu, "Convolutional neural network-based fractional-pixel motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 3, pp. 840–853, Mar. 2018.

[85] Y. Vatis and J. Ostermann, "Adaptive interpolation filter for H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 2, pp. 179–192, Feb. 2009.

[86] L. Zhao, S. Wang, X. Zhang, S. Wang, S. Ma, and W. Gao, "Enhanced CTU-level inter prediction with deep frame rate up-conversion For high efficiency video coding," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Mar. 2018, pp. 206–210.

[87] L. Zhao, "Enhanced motion-compensated video coding with deep virtual reference frame generation," *IEEE Trans. Image Process.*, to be published.

[88] S. Niklaus, L. Mai, and F. Liu. (2017). "Video frame interpolation via adaptive separable convolution." [Online]. Available: https://arxiv.org/abs/1708.01692

[89] J. Chen, E. Alshina, G. J. Sullivan, J.-R. Ohm, and J. Boyce, *Algorithm Description of Joint Exploration Test Model 1*, document JVET-A1001, Oct. 2015, pp. 1–48.

[90] Z. Zhao, S. Wang, S. Wang, X. Zhang, S. Ma, and J. Yang, "CNN-based bi-directional motion compensation for high efficiency video coding," in *Proc. Int. Symp. Circuits Syst. (ISCAS)*, Aug. 2018, pp. 1–4.

[91] Z. Zhao, S. Wang, S. Wang, and J. Yang, "Enhanced Bi-prediction with convolutional neural network for high efficiency video coding," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[92] H. Zhang, L. Song, Z. Luo, and X. Yang, "Learning a convolutional neural network for fractional interpolation in HEVC inter coding," in *Proc. Vis. Commun. Image Process. (VCIP)*, Dec. 2017, pp. 1–4.

[93] N. Yan, D. Liu, H. Li, T. Xu, F. Wu, and B. Li, "Convolutional neural network-based invertible half-pixel interpolation filter for video coding," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Aug. 2018, pp. 201–205.

[94] M. M. Alam, T. D. Nguyen, M. T. Hagan, and D. M. Chandler, "A perceptual quantization strategy for HEVC based on a convolutional neural network trained on natural images," *Proc. SPIE*, vol. 9599, Aug. 2015, Art. no. 959918.

[95] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[96] R. Song, D. Liu, H. Li, and F. Wu, "Neural network-based arithmetic coding of intra prediction modes in HEVC," in *Proc. Vis. Commun. Image Process. (VCIP)*, Feb. 2017, pp. 1–4.

[97] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[98] S. Puri, S. Lasserre, and P. L. Callet, "CNN-based transform index prediction in multiple transforms framework to assist entropy coding," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Feb. 2017, pp. 798–802.

[99] G. Cote, B. Erol, M. Gallant, and F. Kossentini, "H.263+: Video coding at low bit rates," *IEEE Trans. Circuits Syst. for Video Technol.*, vol. 8, no. 7, pp. 849–866, Nov. 1998.

[100] Y. Zhang, T. Shen, X. Ji, Y. Zhang, R. Xiong, and Q. Dai, "Residual highway convolutional neural networks for in-loop filtering in HEVC," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3827–3241, Aug. 2018.

[101] C. Jia, S. Wang, X. Zhang, S. Wang, and S. Ma, "Spatial-temporal residue network based in-loop filter for video coding," in *Proc. Vis. Commun. Image Process. (VCIP)*, Mar. 2017, pp. 1–4.

[102] C. Jia *et al.*, "Content-aware convolutional neural network for in-loop filtering in high efficiency video coding," *IEEE Trans. Image Process.*, to be published.

[103] X. F. Zhang, S. Wang, K. Gu, W. Lin, S. Ma, and W. Gao, "Just-noticeable difference-based perceptual optimization for JPEG compression," *IEEE Signal Process. Lett.*, vol. 24, no. 1, pp. 96–100, Jan. 2017.

[104] X. Song *et al.* (2018). "A practical convolutional neural network as loop filter for intra frame." [Online]. Available: https://arxiv.org/abs/1805.06121

[105] W.-S. Park and M. Kim, "CNN-based in-loop filtering for coding efficiency improvement," in *Proc. IEEE 12th Image, Video, Multidimensional Signal Process. Workshop*, Jun. 2016, pp. 1–5.

[106] J. Kang, S. Kim, and K. M. Lee, "Multi-modal/multi-scale convolutional neural network based in-loop filter design for next generation video codec," in *Proc. Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 26–30.

[107] C. Dong, Y. Deng, C. Change Loy, and X. Tang, "Compression artifacts reduction by a deep convolutional network," in *Proc. IEEE Int. Conf. Comput. Vis.*, May 2015, pp. 576–584.

[108] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 184–199.

[109] K. Li, B. Bare, and B. Yan, "An efficient deep convolutional neural networks model for compressed image deblocking," in *Proc. Int. Conf. Multimedia Expo (ICME)*, Jul. 2017, pp. 1320–1325.

[110] R. Yang, M. Xu, Z. Wang, and T. Li, "Multi-frame quality enhancement for compressed video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Feb. 2018, pp. 6664–6673.

[111] L. Zhu, Y. Zhang, S. Wang, H. Yuan, S. Kwong, and H. H.-S. Ip, "Convolutional neural network-based synthesized view quality enhancement for 3D video coding," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5365–5377, Nov. 2018.

[112] L. Cavigelli, P. Hager, and L. Benini, "CAS-CNN: A deep convolutional neural network for image compression artifact suppression," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 752–759.

[113] B. Zheng, R. Sun, X. Tian, and Y. Chen, "S-Net: a scalable convolutional neural network for JPEG compression artifact reduction," *J. Electron. Imag.*, vol. 27, no. 4, 2018, Art. no. 043037.

[114] T. Chen, H. Liu, Q. Shen, T. Yue, X. Cao, and Z. Ma, "DeepCoder: A deep neural network based video compression," in *Proc. Vis. Commun. Image Process. (VCIP)*, Dec. 2017, pp. 1–4.

[115] Z. Chen, T. He, X. Jin, and F. Wu. (2018). "Learning for video compression." [Online]. Available: https://arxiv.org/abs/1804.09869

[116] M. Ranzato, A. Szlam, J. Bruna, M. Mathieu, R. Collobert, and S. Chopra. (2014). "Video (language) modeling: a baseline for generative models of natural videos." [Online]. Available: https://arxiv.org/abs/1412.6604

[117] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using LSTMS," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 843–852.

[118] Z. Liu, X. Yu, Y. Gao, S. Chen, X. Ji, and D. Wang, "CU partition mode decision for HEVC hardwired intra encoder using convolution neural network," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5088–5103, Nov. 2016.

[119] N. Song, Z. Liu, X. Ji, and D. Wang, "CNN oriented fast PU mode decision for HEVC hardwired intra encoder," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Feb. 2017, pp. 239–243.

[120] M. Xu, T. Li, Z. Wang, X. Deng, R. Yang, and Z. Guan, "Reducing complexity of HEVC: A deep learning approach," *IEEE Trans. Image Process.*, to be published.

[121] X. Zhang, S. Ma, S. Wang, X. Zhang, H. Sun, and W. Gao, "A joint compression scheme of video feature descriptors and visual content," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 633–647, Feb. 2017.

[122] Y. Li, C. Jia, X. Zhang, S. Wang, S. Ma, and W. Gao, "Joint rate-distortion optimization for simultaneous texture and deep feature compression of facial images," in *Proc. IEEE Int. Conf. Multimedia Big Data (BigMM)*, Aug. 2018, pp. 334–341.

[123] L. Su, Y. Lu, F. Wu, S. Li, and W. Gao, "Complexity-constrained H.264 video encoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 4, pp. 477–490, Apr. 2009.

Siwei Ma (M'12) received the B.S. degree from Shandong Normal University, Jinan, China, in 1999, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2005. He held a post-doctoral position with the University of Southern California at Los Angeles, Los Angeles, CA, USA, from 2005 to 2007. He joined the School of Electronics Engineering and Computer Science, Institute of Digital Media, Peking University, Beijing, where he is currently a Professor. He has authored more than 200 technical articles in refereed journals and proceedings in image and video coding, video processing, video streaming, and transmission. He is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and the *Journal of Visual Communication and Image Representation*.

Xinfeng Zhang (M'16) received the B.S. degree in computer science from the Hebei University of Technology, Tianjin, China, in 2007, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2014. From 2014 to 2017, he was a Research Fellow with the Rapid-Rich Object Search Lab, Nanyang Technological University, Singapore. From 2017 to 2018, he was a Post-Doctoral Fellow with the School of Electrical Engineering System, University of Southern California, Los Angeles, CA, USA. From 2018 to 2019, he was a Research Fellow with the Department of Computer Science, City University of Hong Kong. He is currently an Assistant Professor with the Department of Computer Science, University of Chinese Academy of Sciences. He has given more than 20 technical proposals to ISO/MPEG, ITU-T, and AVS standards, and has authored more than 100 refereed journals/conference papers. His research interests include video compression, image/video quality assessment, and image/video analysis. He received the Best Paper Award at the 2017 Pacific-Rim Conference on Multimedia, the Best Paper Award of IEEE Multimedia 2018, and is the coauthor of a paper that received the Best Student Paper Award in the IEEE International Conference on Image Processing 2018.

Chuanmin Jia (S'18) received B.E. degree in computer science from the Beijing University of Posts and Telecommunications, Beijing, China, in 2015. He is currently pursuing the Ph.D. degree with the Department of Computer Science, Peking University, Beijing, China. He was a Visiting Student with the Video Lab, New York University, NY, USA, in 2018. His research interests include video compression, light field compression, and machine learning.

He received the Best Paper Award of the Pacific-Rim Conference on Multimedia in 2017 and the IEEE Multimedia in 2018, and the Best Student Paper Award in the IEEE International Conference on Multimedia Information Processing and Retrieval in 2019.

Zhenghui Zhao received the B.S. degree from the School of Mathematical Sciences, Dalian University of Technology, in 2015. He is currently pursuing the Ph.D. degree with Peking University. His research interests include image and video compression, and image and video processing.

Shiqi Wang received the B.S. degree in computer science from the Harbin Institute of Technology in 2008 and the Ph.D. degree in computer application technology from Peking University in 2014. From 2014 to 2016, he was a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada. From 2016 to 2017, he was with the Rapid-Rich Object Search Laboratory, Nanyang Technological University, Singapore, as a Research Fellow. He is currently an Assistant Professor with the Department of Computer Science, City University of Hong Kong. He has given more than 40 technical proposals to ISO/MPEG, ITU-T, and AVS standards, and has authored more than 150 refereed journals/conference papers. His research interests include video compression, image/video quality assessment, and image/video search and analysis. He received the Best Paper Award at the 2017 Pacific-Rim Conference on Multimedia, the Best Paper Award of IEEE Multimedia 2018, and is the coauthor of a paper that received the Best Student Paper Award in the IEEE International Conference on Image Processing 2018.

Shanshe Wang received the B.S. degree from the Department of Mathematics, Heilongjiang University, Harbin, China, in 2004, the M.S. degree in computer software and theory from Northeast Petroleum University, Daqing, China, in 2010, and the Ph.D. degree in computer science from the Harbin Institute of Technology. He held a post-doctoral position with Peking University from 2016 to 2018. He joined the School of Electronics Engineering and Computer Science, Institute of Digital Media, Peking University, Beijing, where he is currently a Research Assistant Professor. His current research interests include video compression and image and video quality assessment.