IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 2, 2020, accepted December 16, 2020, date of publication December 21, 2020,
date of current version January 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3046040

Deep Predictive Video Compression Using Mode-Selective Uni- and Bi-Directional Predictions Based on Multi-Frame Hypothesis

WOONSUNG PARK AND MUNCHURL KIM, (Senior Member, IEEE)
Korea Advanced Institute of Science and Technology, Daejeon 305-701, South Korea

Corresponding author: Munchurl Kim (mkimee@kaist.ac.kr)

This work was fully supported by the Institute for Information and Communications Technology Promotion (IITP) Grant funded by the Ministry for Science and ICT (Information and Communication Technology) of the Korean Government, Intelligent High Realistic Visual Processing for Smart Broadcasting Media, under Grant 2017-0-00419. This was also partially supported by the BK21 (Brain Korea 21) Program.

ABSTRACT Recently, deep learning-based image compression has shown significant performance improvement in terms of coding efficiency and subjective quality. However, there has been relatively less effort on video compression based on deep neural networks. In this paper, we propose an end-to-end deep predictive video compression network, called DeepPVCnet, using mode-selective uni- and bi-directional predictions based on multi-frame hypothesis with a multi-scale structure and a temporal-context-adaptive entropy model. Our DeepPVCnet jointly compresses motion information and residual data that are generated from the multi-scale structure via the feature transformation layers. Recent deep learning-based video compression methods were proposed in a limited compression environment using only P-frame or B-frame. Learned from the lesson of the conventional video codecs, we firstly incorporate a mode-selective framework into our DeepPVCnet with uni- and bi-directional predictive modes in a rate-distortion minimization sense. Also, we propose a temporal-context-adaptive entropy model that utilizes the temporal context information of the reference frames for the current frame coding. The autoregressive entropy models for CNN-based image and video compression is difficult to compute with parallel processing. On the other hand, our temporal-context-adaptive entropy model utilizes temporally coherent context from the reference frames, so that the context information can be computed in parallel, which is computationally and architecturally advantageous. Extensive experiments show that our DeepPVCnet outperforms AVC/H.264, HEVC/H.265 and state-of-the-art methods in an MS-SSIM perspective.

INDEX TERMS AVC/H.264, deep learning, frame prediction, HEVC/H.265, and video compression.

I. INTRODUCTION

Conventional video codecs such as AVC/H.264 [45], HEVC/H.265 [38] and VP9 [29] have shown significantly improved coding efficiencies, especially by enhancing their temporal prediction accuracies for the current frame to be encoded using its adjacent frames. In particular, there are three coding modes of frames used in video compression: I-frame (intra-coded frame) mode that is compressed independently from its adjacent frames; P-frame mode that is compressed through the forward prediction using motion information; and B-frame mode that is compressed with bi-directional prediction for the current frame. P-frame coding is suitable for low latency in video compression. In perceptive of coding efficiency, B-frame coding provides the highest coding efficiency compared to the I-frame and P-frame coding. Therefore, the standard codecs [38], [45] use both P-frame and B-frame coding methods for video coding.

Deep learning-based approaches have recently shown significant performance improvement in image processing. Especially, in the field of low-level computer vision, intensive research has been made for deep learning-based image super-resolution [12], [18], [20], [24] and frame interpolation [15], [28], [30]–[32]. In addition, there are many recent studies on image compression using deep learning [5], [6], [16], [21], [23], [27], [35], [40]–[42] which often incorporate auto-encoder based end-to-end image compression architectures by attempting to improve compression performance. These works showed outperformed results of coding efficiency compared to the traditional image compression methods such as JPEG [43], JPEG2000 [37], and BPG [7]. While the image compression tries to reduce only

The associate editor coordinating the review of this manuscript and approving it for publication was Hualong Yu.

spatial redundancy around the neighboring pixels with limited coding efficiency, traditional video compression can achieve significant compression performance because it can take advantage of temporal redundancy among neighboring frames. Also, by exploiting the temporal redundancy, deep learning-based video compression has been studied in two main directions: First, some components (or coding tools) in the conventional video codecs are replaced with deep neural networks. For example, Park and Kim [33] first tried to improve compression performance by replacing the in-loop filters of HEVC with a CNN-based in-loop filer. In [10], Cui *et al.* proposed intra-prediction method with CNN in HEVC to improve compression performance. In [51], Zhao *et al.* replaced the bi-prediction strategy in HEVC with CNN to improve coding efficiency; Second, there are studies to improve the compression performance by using auto-encoder based end-to-end neural network architectures [4], [9], [11], [13], [25], [26], [36], [46], [47]. Although deep learning-based image compression has been intensively studied, video compression has drawn less attention. In this paper, we propose an end-to-end deep predictive video compression network, called DeepPVCnet, using mode-selective uni- and bi-directional predictions based on multi-frame hypothesis with a multi-scale structure and a temporal-context-adaptive entropy model. The contributions of our proposed DeepPVCnet are as follows:

- We first show a mode-selective framework with both uni- and bi-directional predictive coding structures for deep learning-based predictive video compression in the rate-distortion minimization sense, thus achieving the improved coding efficiency. The selected mode information for frame prediction is transmitted to decoder sides with a negligible amount of bits;
- We propose a temporal-context-adaptive entropy model that utilizes temporally coherent context information from the multiple reference frames to estimate the parameters of Gaussian entropy models for the quantized latent representation of the current frame. While the autoregressive entropy models for CNN-based image compression suffer from serialized processing, our temporal-context-adaptive entropy model allows for context computation in parallel;
- Our DeepPVCnet tries to jointly compresses motion and residual information based on a multi-scale structure for the current frame and its reference frames via learned *feature transformation* in encoder sides. This structure can effectively reduce the coupled redundancy of motion and residual information;
- Contrary to the deep neural network-based state-of-the-art (SOTA) methods [26], [46] that reply on a single reference frame for each prediction direction, our method improves prediction accuracy for the current frame by utilizing multiple frames for both uni- and bi-directional prediction modes.

This paper is organized as follows: Section II introduces the related work with deep neural network-based image/video compression, optical flow estimation and frame interpolation; In Section III, we introduce the details of our proposed deep video compression network, called DeepPVCnet; Section IV presents the experimental results to show the effectiveness of our DeepPVCnet compared to the conventional video codecs and SOTA methods [4], [13], [25], [26], [46], [47]; Finally, we conclude our work in Section V.

II. RELATED WORK

Both conventional image compression (such as JPEG, JPEG2000, and BPG) and video compression (AVC/H.264, HEVC, and VP9) methods have shown high compression performance. Recently, deep learning-based image and video compression methods have been actively studied. The key element that brings up high coding efficiency in video coding is temporal prediction to reduce temporal redundancy. Therefore, we also review deep learning-based optical flow estimation and frame interpolation networks that are essential elements for predictive coding.

Deep Learning-Based Image Compression: Unlike conventional image compression based on transform coding, recent deep learning-based image compression methods often adopt auto-encoder structures that perform nonlinear transforms. First, there are several works on image compression using Long Short Term Memory (LSTM)-based auto-encoders [16], [41], [42] where a progressive coding concept is used to encode the difference between the original image and the reconstructed image. In addition, there are studies on image compression using convolutional neural network (CNN) based auto-encoder structures by modeling the feature maps of the bottleneck layers for entropy coding [5], [6], [21], [23], [27], [35], [40]. In [6], Ballé *et al.* introduced an input-adaptive entropy model that estimates the scales of the latent representations depending on the input. In [21], Lee *et al.* have proposed a context-adaptive entropy model for image compression which uses two types of contexts: bit-consuming context and bit-free context. Their models in [6], [21] outperformed the conventional image codecs such as BPG. Our DeepPVCnet also adopts such an auto-encoder structure used in [6] as the baseline structure combined with our temporal-context-adaptive entropy model.

Deep Learning-Based Video Compression: There are two main directions of deep learning based video compression research: The first is to replace the existing components of the conventional video codecs with deep neural networks (DNN). For example, there are some works to replace in-loop filters with deep neural networks [14], [17], [33], [49], and post-processing to enhance the resulting frames of the conventional video codecs [22], [48]. The intra/inter predictive coding modules have also been substituted with DNN modules for video coding [10], [33]; And, the second direction includes CNN-based auto-encoder structures without the coding tools of conventional video codecs involved. In [26], Lu *et al.* proposed the first end-to-end deep video compression network that jointly optimizes

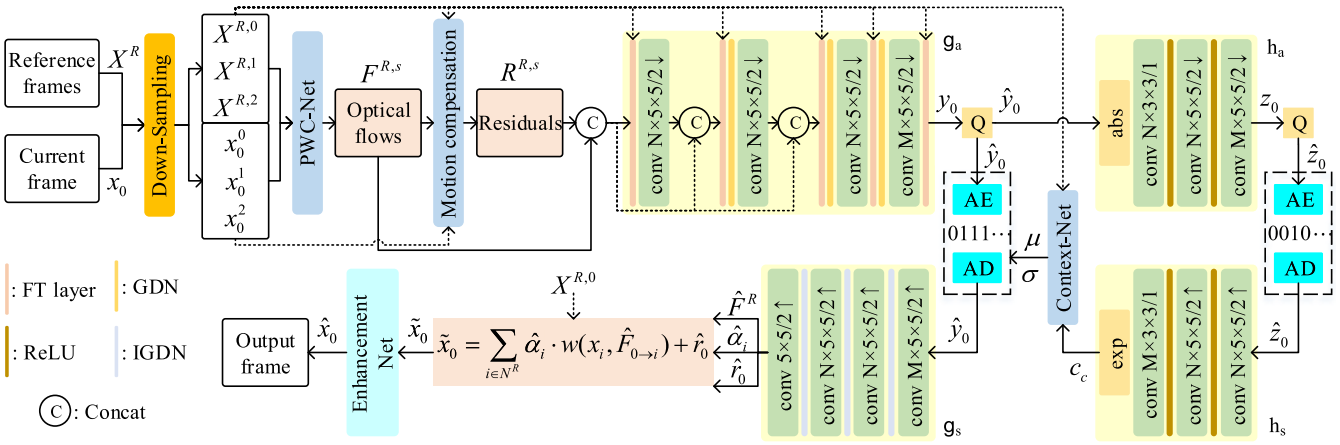

FIGURE 1. Overall architecture of our proposed DeepPVCnet. The notations of the parameters of the convolutional layers are denoted as: number of filters × filter height × filter width / the up- or down-scale factor, where ↑ and ↓ denote the up- and down-scaling, respectively. s denotes bilinear down-sampling factor for reference frames X^R and current frame x_0. The 'Motion compensation' process is performed by Eq. 2. AE and AD represent an arithmetic encoder and an arithmetic decoder, respectively. Also, Q represents a quantizer for latent representations y_0.

all the components for low-latency scenarios of video compression. Then, Lin *et al.* in [25] extended it by utilizing multiple reference frames for low-latency scenarios of video compression. In [36], Rippel *et al.* proposed a novel video compression framework with propagation of the learned state and ML-based spatial rate control. In [4], Agustsson *et al.* proposed a low-latency video compression model based on the scale-space flow for better handling disocclusions and fast motion. However, the methods in [4], [25], [26], [36] have been proposed for P-frame predictive coding which only use previously encoded frames to predict the current frame. In [46], Wu *et al.* proposed a ConvLSTM-based video compression method to improve the coding efficiency. However, this method used conventional block motion estimation and compression methods, which will degrade the coding efficiency. In [9], Cheng *et al.* proposed a frame interpolation network based deep video compression. Since this work utilized a pre-trained frame interpolation network [32] from two frames which are far from each other, the prediction performance will be significantly reduced. The method in [11] uses both an interpolation- and residual-based auto-encoder for B-frame coding. It shows good performance in high bit ranges, but not in low or mid bit ranges. In [47], Yang *et al.* proposed a hierarchical learned video compression method with the hierarchical quality layers and a recurrent enhancement network. The method in [13] incorporates a 3D auto-encoder that does not use the P-frame and B-frame coding concept. However, it requires lots of memory and computational complexity. In contrast to the SOTA methods using a single frame as reference for prediction, our method utilizes multiple frames as reference to predict the current frame. While these SOTA methods take either a P-frame or a B-frame coding structure for video compression, in this paper, we propose a mode-selective framework with uni- and bi-directional prediction modes where the best mode is selected in a rate-distortion optimization sense and is signaled to the decoder side.

Deep Learning-Based Optical Flow Estimation and Frame Interpolation: Optical flow estimation and frame interpolation can be used for predictive video coding. There have been many studies related to optical flow estimation using deep neural networks. In [34], Ranjan *et al.* introduced SpyNet that uses a spatial pyramid network and warps the second image to the first image with the initial optical flow. Also, the PWC-Net [39] was introduced with a learnable feature pyramid structure that uses the estimated current optical flow to warp the CNN feature maps of the second image. Their model outperformed all previous optical flow methods. Since they use the feature pyramid structures, the optical flow estimation is robust to large motion over other deep neural network-based optical flow methods. Therefore, we incorporate the pre-trained PWC-Net as an initial parameters of the optical flow estimation network into our DeepPVCnet. Recent CNN-based frame interpolation methods include convolution filtering-based [31], [32], phase-based [28], and optical flow-based [15], [30] approaches. The convolution filtering-based frame interpolation is based on the frame prediction between adjacent frames through convolution filtering operation without using optical flow. The phase-based frame interpolation uses a CNN to reduce the reconstruction loss in the phase domain rather than in the image domain. Finally, the optical flow-based interpolation generates the frames between two frames through a CNN after warping with optical flow between two frames. In this paper, we adopt an optical flow-based prediction scheme [15] for our DeepPVCnet.

III. PROPOSED METHOD

Our proposed DeepPVCnet for both uni- and bi-directional predictive coding is illustrated in Fig. 1. As depicted in Fig. 1, our DeepPVCnet uses the neighboring frames as reference frames to compress the current frame. The reference frames, denoted as X^R, for current frame x_0 are composed of $\{x_{-2}, x_{-1}\}$ and $\{x_{-2}, x_{-1}, x_1, x_2\}$ for uni- and bi-directional

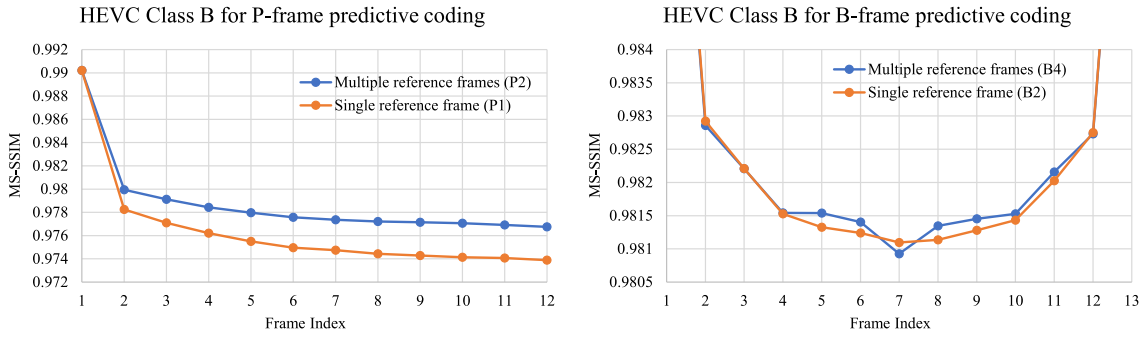

FIGURE 2. Compression performance comparison for a single reference frame and multiple reference frames. The bitrates of the P-frame coding models are about 0.37 bpp and those of the B-frame coding models are about 0.59 bpp for HEVC Class B dataset [38].

coding, respectively. For X^R and x_0, the bilinear downsampling with a scale index s is performed for multi-scale motion estimation and compensation as follows:

$$X^{R,s} = \{Down(x, s) \,|\, x \in X^R\}$$
$$x_0^s = Down(x_0, s) \tag{1}$$

where $X^{R,s}$ and x_0^s denotes the down-scaled reference frames and the down-scaled current frame with the scale index s, respectively. $Down(\cdot, s)$ denotes a bilinear downsampling process with a scale factor 2^s ($s = 0, 1, 2$ for our experiments).

Each reference frame in $X^{R,s}$ is concatenated to x_0^s for estimating the optical flow between these frames using the fine-tuned PWC-Net [39]. The resulting optical flows $F^{R,s}$ are composed of $\{F^s_{0\to-2}, F^s_{0\to-1}\}$ and $\{F^s_{0\to-2}, F^s_{0\to-1}, F^s_{0\to1}, F^s_{0\to2}\}$ for uni- and bi-directional coding, respectively. Then, the prediction frames $P^{R,s}$ are calculated by a backward warping function $w(\cdot,\cdot)$ [15] with $X^{R,s}$ and $F^{R,s}$. The resulting prediction frames $P^{R,s}$ are composed of $\{p^s_{0\leftarrow-2}, p^s_{0\leftarrow-1}\}$ and $\{p^s_{0\leftarrow-2}, p^s_{0\leftarrow-1}, p^s_{0\leftarrow1}, p^s_{0\leftarrow2}\}$ for uni- and bi-directional coding, respectively. The residual frames $R^{R,s}$ can be expressed as follows:

$$R^{R,s} = \{x_0^s - p^s_{0\leftarrow k} \,|\, p^s_{0\leftarrow k} \in P^{R,s}\} \tag{2}$$

where k denote an relative index of the reference frame for the current frame x_0^s. $R^{R,s}$ are composed of $\{r^s_{0\leftarrow-2}, r^s_{0\leftarrow-1}\}$ and $\{r^s_{0\leftarrow-2}, r^s_{0\leftarrow-1}, r^s_{0\leftarrow1}, r^s_{0\leftarrow2}\}$ for uni- and bi-directional coding, respectively.

The joint information of $F^{R,0}$ and $R^{R,0}$ for scale 0 is mapped to a latent representation y_0 through the encoder network g_a with five feature transformation (FT) layers. Similarly, the joint information of $F^{R,s}$ and $R^{R,s}$ for scales $s = 1, 2$ are concatenated into the feature maps of the same sizes in the encoder network g_a as depicted in Fig. 1.

After the quantization step, we can obtain the quantized latent representation $\hat{y}_0$. Then, the reconstructed optical flows $\hat{F}^R$, the reconstructed residual frame $\hat{r}_0$ and the synthesis coefficients $\hat{\alpha}_i$ are estimated by the decoder network g_s with the entropy model of $\hat{y}_0$. The reconstructed frame $\tilde{x}_0$ is given by

$$\tilde{x}_0 = \sum_{i \in N^R} \hat{\alpha}_i \cdot w(x_i, \hat{F}_{0\to i}) + \hat{r}_0 \tag{3}$$

where the set, N^R, of reference frame indices are composed of $\{-2, -1\}$ and $\{-2, -1, 1, 2\}$ for uni- and bi-directional predictive coding, respectively. $\hat{F}^R$ consist of $\{\hat{F}_{0\to-2}, \hat{F}_{0\to-1}\}$ and $\{\hat{F}_{0\to-2}, \hat{F}_{0\to-1}, \hat{F}_{0\to1}, \hat{F}_{0\to2}\}$ for uni- and bi-directional predictive coding, respectively. Finally, the Enhancement Net outputs enhanced frame $\hat{x}_0$ from $\tilde{x}_0$. The details of the proposed network are described in Section III.A-III.E.

A. MULTIPLE REFERENCE FRAMES

In general, the conventional video codecs like AVC/H.264 and HEVC/H.265 compress the current frame using multiple reference frames for each prediction direction. The usage of multiple reference frames allows to effectively deal with occlusion problems, thus resulting in accurate prediction for the current frame. In video compression, the quantization errors are propagated as subsequent frames are compressed. By using multiple reference frames, such a quantization error propagation can be alleviated for the prediction of the current frame, thus increasing the prediction accuracy and coding efficiency. By compromising the complexity of incorporating multiple reference frames, our DeepPVCnet utilizes two reference frames for uni-directional predictive coding (P2 in Fig. 2) and four reference frames for bi-directional predictive coding (B4 in Fig. 2) where two for forward and the other two for backward prediction, in contrast to the state-of-the-art methods [9], [11], [26], [46] of deep learning-based video compression. The effectiveness of using multiple reference frames is shown in Fig. 2.

B. COMPRESSING JOINT INFORMATION WITH FEATURE TRANSFORMATION (FT) LAYERS

We incorporate five feature transformation (FT) layers into the encoder side of the CNN-based auto-encoder structure g_a that jointly compresses the multi-scale motion information and residuals. To cope with various amounts of motion for different video sequences, multi-scale motion estimation and compensation are performed at the encoder side. Then, the generated multi-scale motions and residuals are concatenated to the output feature maps of each convolution layer, which are then fed as input into the following FT layer of the encoder network. By doing so, the multi-scale joint information of motions and residuals can be effectively fused for better compression.

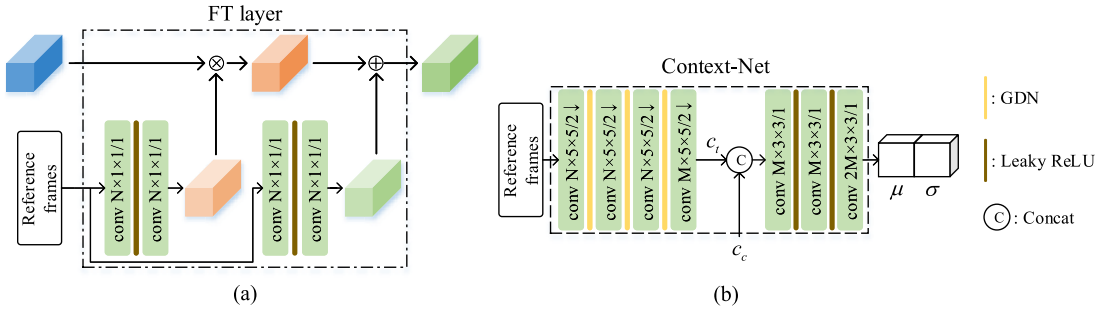

FIGURE 3. The feature transformation (FT) layers and the Context-Net for our DeepPVCnet: (a) The FT layers includes pixel-wise multiplication and addition parameters. (b) The Context-Net utilizes two context information c_c and c_t for estimating the parameters of Gaussian entropy model of $\hat{y}_0$.

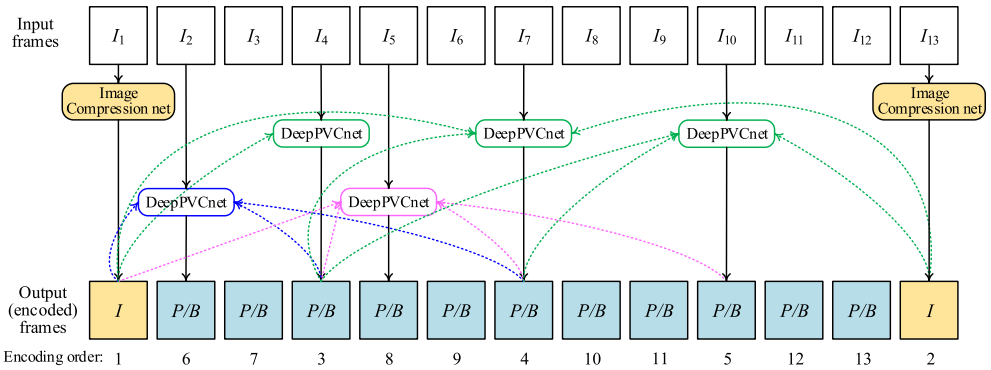

FIGURE 4. GOP structure for our mode-selective framework with uni- and bi-directional predictions.

The recent methods [9], [11], [26], [46] in deep video compression are designed to compress single-scale optical flow and residual separately, but our proposed DeepPVCnet jointly compresses multi-scale motion information and residual with compactization towards improving coding efficiency under the assumption that the redundancy between motion information and residual exists. Also, the FT layers alter interim feature output by the learned transformation under the guidance of multiple reference frames, which can help reducing the coupled redundancy between the multi-scale motion information and residual. The details of the FT layers are depicted in Fig. 3-(a). As shown in Fig. 3-(a), the FT layers serve to perform affine transform of each element of the input feature map. The parameters of the affine transform are learned with respect to the reference frames via two convolutional layers.

C. TEMPORAL-CONTEXT-ADAPTIVE ENTROPY MODEL

We propose a temporal-context-adaptive entropy model for the quantized latent representation $\hat{y}_0$. Our proposed entropy model adopts the basic structure [6] with the hyperprior $\hat{z}_0$ and the hyper encoder-decoder network pair (h_a, h_s) as shown in Fig. 1. The output feature map of the hyper encoder-decoder network is the context information c_c of current frame x_0. Since there exists a contextual similarity between x_0 and X^R, we propose a Context-Net to estimate the mean μ and standard deviation σ of a Gaussian model for $\hat{y}_0$ as follows:

$$\begin{aligned} \hat{y}_0 &\sim \mathcal{N}(\mu, \sigma^2) \\ (\mu, \sigma) &= \text{Context-Net}(X^R, c_c) \end{aligned} \tag{4}$$

where our proposed Context-Net extracts the context information c_c of x_0 and the temporal context information c_t of X^R. Then, it concatenates c_c and c_t to obtain the μ and σ as the same spatial size as $\hat{y}_0$. The Context-Net is illustrated in Fig. 3-(b). As shown in Fig. 3-(b), c_t is generated using the reference frames via four convolutional layers. Then, μ and σ are estimated for c_t and c_c via three convolutional layers. More details of the entropy coding model are represented in Appendix B.

D. MODE-SELECTIVE FRAMEWORK

The SOTA deep learning-based video compression methods [4], [9], [11], [25], [26], [36], [46] tend to have a limitation that only compresses all of the frames in either a P-frame or a B-frame coding structure. Fig. 4 depicts a GOP (Group of Pictures) structure for our mode-selective framework with uni- or bi-directional predictions in a similar way as traditional video codecs. In our mode-selective framework, each frame can be encoded with an intra-mode, a uni-directional prediction mode or a bi-directional prediction mode. The uni-directional prediction mode has two sub-modes: M^f_{uni} for forward prediction and M^b_{uni} for backward prediction, and the bi-directional prediction mode is denoted as M_{bi}. For the GOP structure in Fig. 4, I_1 and I_{13} are encoded as the intra-mode using a pre-trained image compression network [21] while all other frames between I_1 and I_{13} are encoded in either M^f_{uni}, M^b_{uni} or M_{bi}. It should be noted in Fig. 4 that the frame I_4 is encoded by referencing I_1 which is encoded *a priori*. Next, I_7 is compressed, followed by I_{10}. Depending on the availability of neighboring encoded frames, one or two encoded frames

TABLE 1. Coding order and reference frames of our method in a test phase.

Frame index (Coding order)	I/P/B	Reference frames (previous)	Reference frames (future)
1	I	-	-
13	I	-	-
4	P	1	-
7	P/B	1, 4	13
10	P/B	4, 7	13
2	P/B	1	4, 7
3	P/B	1	4, 7
5	P/B	1, 4	7, 10
6	P/B	1, 4	7, 10
8	P/B	4, 7	10, 13
9	P/B	4, 7	10, 13
11	P/B	7, 10	13
12	P/B	7, 10	13

are referenced to encoded each frame between I_1 and I_{13} as shown in Fig. 4. I_3 and I_6 may use the same reference frames as I_2 and I_5, respectively. I_8 and I_9 have the same referencing structures as I_6 and I_5, respectively. Also, I_{11} and I_{12} have the same referencing structures as I_3 and I_2, respectively. The details of the coding order and the selection rule of the reference frame for our proposed method in a test phase are represented in Table 1. Note that for frames in which the reference frames are not multiple, duplicated reference frames are utilized as the inputs of deepPVCnet in our experiment.

The selected mode information is sent to the decoder sides as two-bit data which is a negligible bit amount. Based on this mode-selective framework, we train each deepPVCnet for M^f_{uni} and M_{bi} where the DeepPVCnet trained for M^f_{uni} is also used for M^b_{uni} by changing the reference frame order. The mode selection can be determined by:

$$\tilde{m} = \underset{m\in\{M^f_{uni},M^b_{uni},M_{bi}\}}{\arg\min} \quad R_{n,p} + \lambda_m \left\| I_n - \hat{I}_{n,m} \right\|^2_F \tag{5}$$

where $R_{n,m}$ and $\hat{I}_{n,m}$ denote the bitrate and the reconstructed frame of I_n with mode m, respectively.

E. ENHANCEMENT NET

To further improve the qualities of the reconstructed frames, the Enhancement Net shown in Fig. 1 is incorporated into the decoder side of our DeepPVCnet to enable the role of an in-loop filter as in the traditional video codecs. We utilize the residual dense network (RDN) [50] which consists of five residual dense blocks (RDB) with three convolution filters per each block for our Enhancement Net which is described in details in Appendix C.

IV. EXPERIMENTS

A. EXPERIMENTAL CONDITIONS

To show the effectiveness of our DeepPVCnet, extensive experiments are carried out to measure the performance of coding efficiency, and our method is compared with other video coding methods. For intra coding, we used a pre-trained CNN-based image compression model in [21]. For uni- and bi-directional predictive coding, we train our DeepPVCnet models for different bitrate ranges and test the trained models for each bitrate range. Note that we set the GOP size G to 12 for all experiments.

Datasets: We train the DeepPVCnet with the UGC dataset [1]. For pre-processing, we excluded HDR, vertical video, interlaced video and the video that are smaller than 720p from the UGC dataset. The number of frames used for training is about 466K. For evaluation, we test the DeepPVCnet on the raw video datasets such as Ultra Video Group (UVG) [2] and the HEVC Standard Test Sequences (Class B, C, D and E) [38]. The UVG dataset contains seven videos of size 1920×1080. The videos in the HEVC dataset have different sizes depending on their class types.

Implementation: The proposed DeepPVCnet is trained in an end-to-end manner, based on the rate-distortion loss L as:

$$L = \mathbb{E}[-\log_2 p_{\hat{y}_0|\hat{z}_0,X^R}(\hat{y}_0|\hat{z}_0, X^R) - \log_2 p_{\hat{z}_0}(\hat{z}_0) + \lambda \cdot d(x_0, \hat{x}_0)] \tag{6}$$

where λ controls the trade-off between rate and distortion terms, and d is the distortion measure, e.g. (1 - MS-SSIM). In Eq. 6, the first term indicates the conditional entropies of $\hat{y}_0$ given $\hat{z}_0$ and X^R, and the second term is the entropy of $\hat{z}_0$. For several bitrate ranges, we train the DeepPVCnet separately for different values of λ where the number of channels of the convolution filters is N except the convolution layer that has M filters to output the latent representation. We set $N = 128$ and $M = 256$ for three lower bitrates and $N = 192$ and $M = 384$ for two higher bitrates. Our DeepPVCnet is trained from scratch with the fixed PWC-Net [39] for 1M iterations using ADAM [19] with the initial learning rate 0.0001. Then, we fine-tune the PWC-Net with other components of our DeepPVCnet for additional 0.5M iterations. In addition, we used a batch size of 8 and a patch of 256×256 randomly cropped from the 466K training frames extracted from the UGC video dataset.

Evaluation: We measure both distortions and bitrates simultaneously. The multi-scale structural similarity index (MS-SSIM) [44] for an RGB color space, which is known to as a better metric for subjective image quality than PSNR, are used to measure the distortions in our experiments. We use bits per pixel (bpp) to measure the bitrates.

B. EXPERIMENTAL RESULTS

The DeepPVCnet is compared with the conventional video codecs such as AVC/H.264 and HEVC/H.265, as well as three deep learning-based video compression methods in [4], [13], [25], [26], [46], [47]. For fair comparison, the GOP size of the conventional video codecs is fixed to 12. We used the *ffmpeg* coding tool [38] and x265 [3] for H.264 and H.265, respectively. We use several settings of the conventional video codecs where the details of settings are described in details in Appendix A. Fig. 5 shows the rate-distortion (R-D) curves produced by our DeepPVCnet, H.264 and H.265, Wu's method [46], DVC [26], Habibian's method [13], M-LVC [25], Yang's method [47] and Agustsson's method [4] for

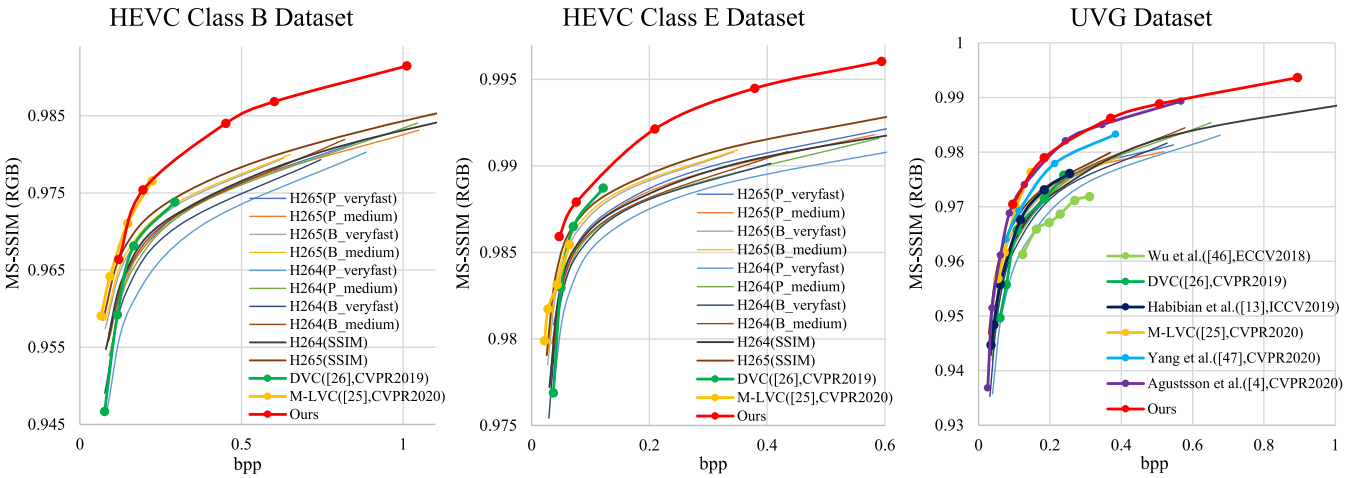

FIGURE 5. MS-SSIM performance comparison of our DeepPVCnet, H.264 [45], H.265 [38], and CNN-based SOTA methods [4], [13], [25], [26], [46], [47] for the UVG dataset, HEVC Class B and E dataset.

TABLE 2. Bjøntegaard Delta bitrate (BD-rate) values with the anchor of H.264 (P_veryfast) with respect to MS-SSIM. Bold values indicate the best results.

Dataset	Sequence name	H.264 (B_medium)	H.265 (B_medium)	Ours
UVG	*Beauty*	-28.11%	-5.52%	**-68.77%**
	Bosphorus	-30.35%	-53.53%	**-69.82%**
	HoneyBee	-31.38%	-54.31%	**-87.68%**
	Jockey	-24.57%	**-47.65%**	-19.41%
	ReadySetGo	-28.30%	**-42.37%**	2.08%
	ShakeNDry	-23.21%	-32.06%	**-77.36%**
	YachtRide	-25.78%	-39.14%	**-59.07%**
	Average	-25.57%	-40.60%	**-57.12%**
HEVC Class B	*BasketballDrive*	-30.41%	-48.50%	**-61.32%**
	BQTerrace	-28.83%	**-40.42%**	-37.01%
	Cactus	-26.76%	-37.68%	**-75.56%**
	Kimono	-28.88%	-42.34%	**-71.65%**
	ParkScene	-34.70%	-45.66%	**-64.84%**
	Average	-29.82%	-42.30%	**-63.19%**
HEVC Class E	*vidyo1*	-20.12%	-46.88%	**-77.44%**
	vidyo3	-26.27%	-39.06%	**-68.40%**
	vidyo4	-17.91%	-39.93%	**-47.91%**
	Average	-21.36%	-41.87%	**-64.98%**

the UVG and HEVC datasets (Class B and E). It can be seen in Fig. 5 that our DeepPVCnet outperforms all the methods over most of the bitrate ranges while other SOTA methods in [13], [25], [26], [46], [47] show the limited results at only low bitrate ranges. In particular, our method shows significantly better compression performance than the other methods for the medium or high bitrate range. More experimental results for the HEVC datasets (Class C and D) and analysis are provided in Appendix D.

Table 2 compares the compression performances of H.264, H.265 and our DeepPVCnet for all test video sequences. In Table 2, we provide the Bjøntegaard Delta bitrate (BD-rate) [8] of the H.264 (B_medium), the H.265 (B_medium) and our DeepPVCnet with the anchor of H.264 (P_veryfast). We calculate the BD-rate values by MS-SSIM where smaller negative values mean that the method uses fewer bits than the anchor. As shown in Table 2, our DeepPVCnet utilizes smaller sizes than H.264 and H.265 to compress the UVG dataset, HEVC Class B and E in average.

C. ABLATION STUDY

For our DeepPVCnet, ablation study is performed for some key components: the multi-scale motion estimation and compensation, the fine-tuned PWC-Net, the multiple reference frames, the temporal-context-adaptive entropy model using the multi-frame hypothesis, mode-selective framework, FT layers and Enhancement Net. In order to demonstrate the contribution of each component, we performed the experiments by excluding the key components one by one from the entire structure of the DeepPVCnet. Fig. 7 represents the resulting MS-SSIM performances of the ablation study.

Multi-Scale

Motion Estimation and Compensation: In order to effectively cope with various motions of different video sequences, we perform motion estimation and compensation based on a multi-scale structure. As can be seen in Fig 7, the multi-scale motion estimation compensation improves the coding gain compared to the single-scale case.

Fine-Tuned

PWC-Net: In [39], the pre-trained PWC-Net has been trained to obtain only high accuracy of optical flows between frames. However, for the video compression problem, the motion estimation network must be trained not only to increase the accuracy of motion estimation, but also to compress the generated motion with high coding efficiency. Therefore, we fine-tuned the PWC-Net to be optimized for video compression in the rate-distortion optimization sense. As shown in Fig. 7, our DeepPVCnet with the fine-tuned PWC-Net outperforms that with the pre-trained PWC-Net for the whole bitrate range.

Multiple Reference Frames: As shown in Figs. 2 and 7, the multiple reference frames contribute to gain high coding efficiency. This gain is achieved thanks to effectively dealing with object occlusions, thus reducing the propagation error. In particular, the multi-frame hypothesis shows better performance in the high bitrate range because our DeepPVCnet can

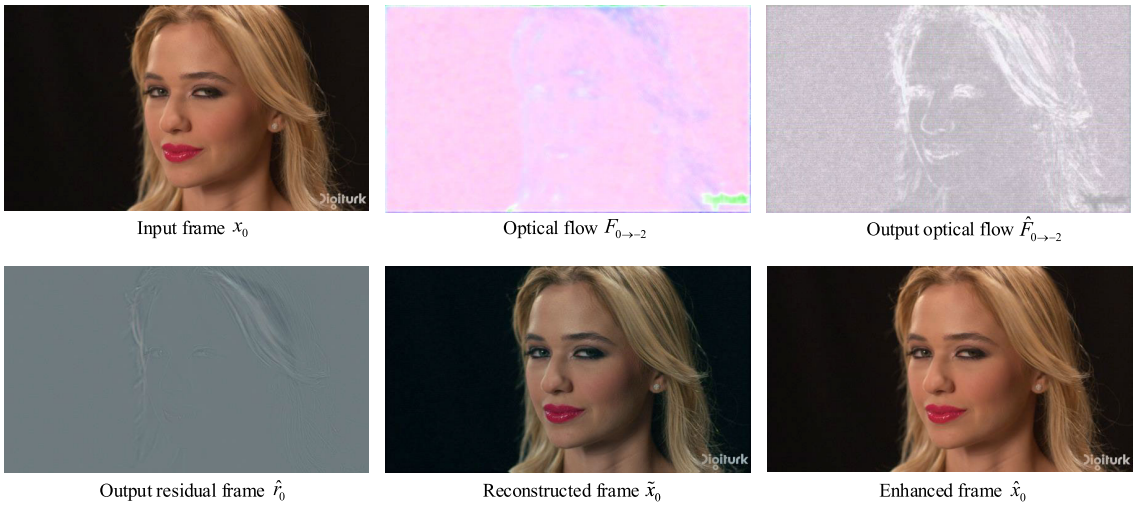

FIGURE 6. Visualization of intermediate feature maps and intermediate outputs of our DeepPVCnet with a B-frame predictive model for *Beauty* sequences of UVG dataset.

TABLE 3. The numbers of parameters for each component of our DeepPVCnet.

	(g_a, g_s)	(h_a, h_s)	Context-Net	FT layers	Enh. Net	PWC-Net [39]	Total
Low	3.4M	3.0M	4.6M	0.12M	0.29M	14.1M	25.5M
High	7.7M	6.8M	10.4M	0.27M	0.29M	14.1M	39.6M

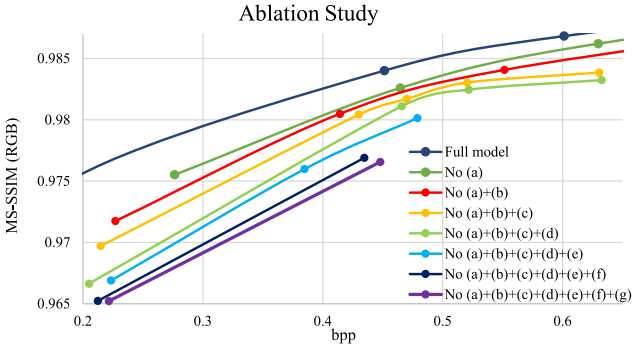

FIGURE 7. Ablation study on the effectiveness of (a) the multi-scale motion estimation and compensation, (b) fine-tuned PWC-Net, (c) multiple reference frames, (d) a mode-selective framework, (e) feature transformation layers, (f) a temporal context-adaptive entropy model, and (g) an Enhancement Net for HEVC Class B dataset. We have the experiments of excluding these components one by one in a row from the DeepPVCnet.

fully utilize neighboring information from multiple reference frames in removing temporal redundancy.

Mode-Selective

Framework: It improves the coding gain especially for low and mid bitrate range as depicted in Fig. 7. Poor prediction in a low bitrate range can be compensated by selectively performing the prediction based on the best prediction mode with our proposed mode-selective framework by Eq. 5. As shown in Fig. 7, our proposed mode-selective framework is a key component to gain high coding efficiency along with the multiple reference frames.

Temporal-Context-Adaptive Entropy Model: As shown in Fig. 7, our DeepPVCnet with the temporal-context-adaptive entropy model achieved coding effiency improvement by reducing the redundancy of the latent representation with the temporal context information of the reference frames. In addition, the proposed entropy model has a structural advantage that it can be computed in parallel in contrast to the autoregressive-based video compression methods [11], [13].

Other Components: The FT layers and an the Enhancement Net have a few parameters compared to those of the entire network. Nevertheless, a slightly improved coding gain has been achieved. In particular, the FT layers allow the encoder to compress joint information effectively.

D. COMPUTATIONAL COMPLEXITY

As shown in Table 3, the total numbers of parameters of our DeepPVCnet are about 25.5M and 39.6M for low and high bitrate models, respectively. For testing, the runtime of our DeepPVCnet was measured in a platform with Intel I9-9900X CPU, 128GB RAM and a single *Titan*^{TM} RTX GPU. For sequences of sizes 416×240, 832×480, 1280×720 and 1920×1080, the encoding and decoding speeds of our DeepPVCnet are (5.9 fps, 44.2fps), (3.9 fps, 15.0fps), (2.2 fps, 6.7fps) and (1.1 fps, 3.2fps), respectively. Especially, the decoding speed is considerably faster than other autoregressive based entropy coding model methods [11], [13]. This is because parallel processing is not possible on the decoder side for these methods [11], [13].

E. VISUAL COMPARISONS

In this section, we visualize the interim results by our DeepPVCnet. Then, we visualize the pre-trained and fine-tuned optical flows by PWC-Net. Also, some reconstructed frames by the H.264, H.265 and our DeepPVCnet are presented for subjective comparison.

Visualization of Feature Maps and Reconstructed Frames: Fig. 6 visualizes the optical flow maps, the output residual frame, a reconstructed frame and an enhanced frame for an

FIGURE 8. Visual comparison of optical flows from a pre-trained PWC-Net and a fine-tuned PWC-Net. The pre-trained PWC-Net generates a lot of smooth area of optical flows because it is trained to only accurately obtain motion between frames, not the direction in which the frame is compressed well. However, since the fine-tuned PWC-Net is trained in the direction in which the frame is well compressed, it also generates the optical flows with the texture area.

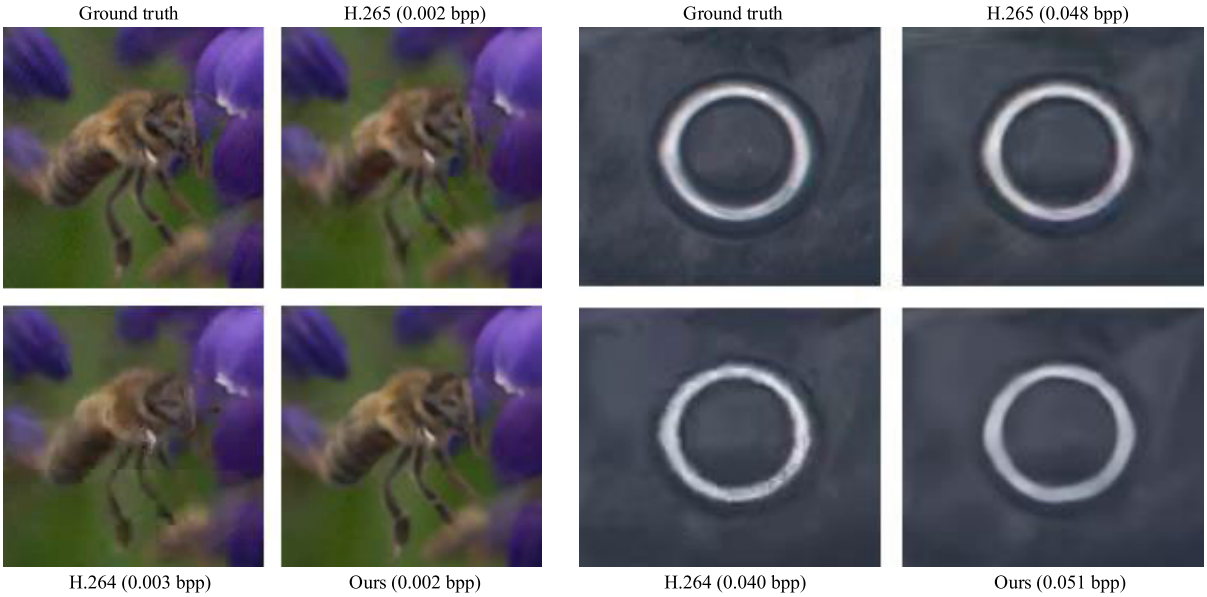

FIGURE 9. Visual comparison on the compression results for cropped frames of a *HoneyBee* sequence (left) obtained by H.264 (0.003 bpp), H.265 (0.002 bpp) and our method (0.002 bpp) and cropped frames of a *YachtRide* sequence (right) obtained by H.264 (0.040 bpp), H.265 (0.048 bpp) and our method (0.051 bpp). (Best viewed in screen).

input frame of a *Beauty* sequence obtained via the pipeline of our DeepPVCnet. The optical flow $F_{0\to-2}$ in Fig. 6 is an input to the encoder network of the DeepPVCnet, which is obtained from the PWC-Net. $\hat{F}_{0\to-2}$ is the output optical flow of the decoder network, which is used to synthesize the current input frame for reconstruction. The output residual frame $\hat{r}_0$ is the difference between the output $\tilde{x}_0$ and the blended output ($\tilde{x}_0 - \hat{r}_0$ in Eq. 3) of warped frames, as shown in Fig. 1.

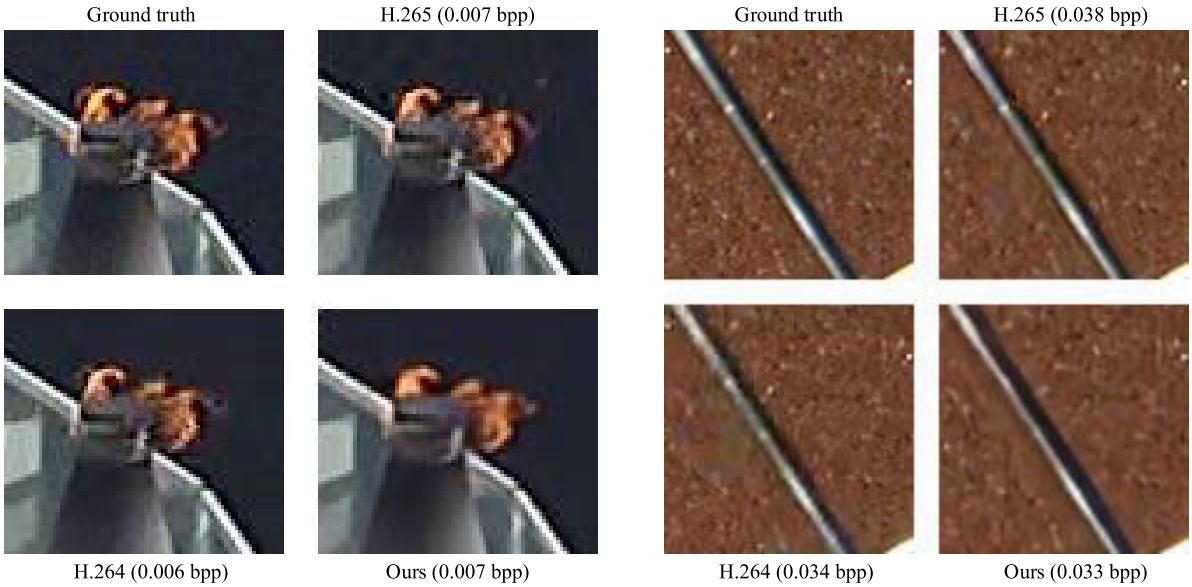

FIGURE 10. Visual comparison on the compression results for cropped *BQTerrace* sequences (left) obtained by H.264 (0.006 bpp), H.265 (0.007 bpp) and our method (0.007 bpp) and cropped *Cactus* sequences (right) obtained by H.264 (0.034 bpp), H.265 (0.038 bpp) and our method (0.033 bpp). (Best viewed in screen).

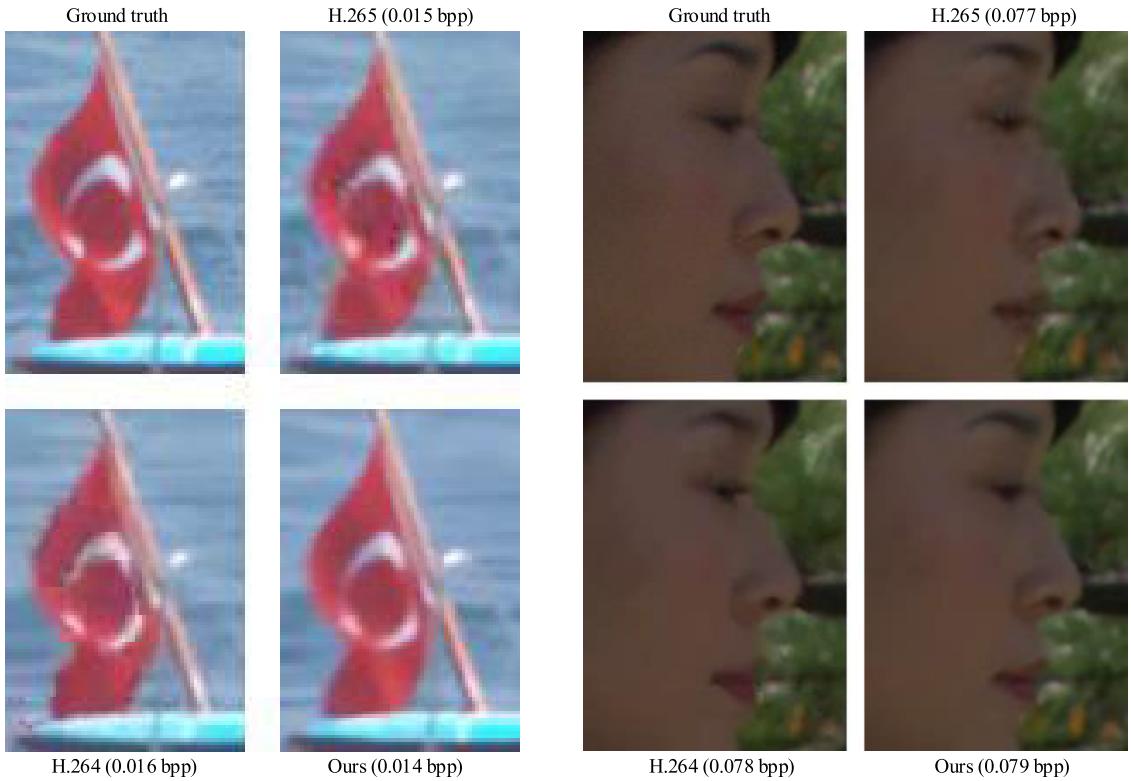

FIGURE 11. Visual comparison on the compression results for cropped frames of a *Bosphorus* sequence (left) obtained by H.264 (0.016 bpp), H.265 (0.015 bpp) and our method (0.014 bpp) and cropped frames of a *Kimono* sequence (right) obtained by H.264 (0.078 bpp), H.265 (0.077 bpp) and our method (0.079 bpp). (Best viewed in screen).

It is noted in Fig. 6 that the optical flows $F_{0\to-2}$ and $\hat{F}_{0\to-2}$ look significantly different because $\hat{F}_{0\to-2}$ are generated to improve compression efficiency. Then, $\hat{F}_{0\to-2}$ includes more texture parts than $F_{0\to-2}$. Also, the output residual frame $\hat{r}_0$ contains texture parts, which makes it possible to reconstruct the areas that are difficult to recover by optical flows only. Finally, the enhanced frame $\hat{x}_0$ is generated from the reconstructed frame $\tilde{x}_0$ by the Enhancement Net, which is visually much closer to the input frame x_0.

Visualization of Pre-Trained and Fine-Tuned Optical Flows: Fig. 8 presents visual comparison of motion information for the pre-trained PWC-Net and the fine-tuned PWC-Net. As shown in Fig. 8, the motion information from the pre-trained PWC-Net contains large-sized fields of smooth motion since the pre-trained PWC-Net does not consider compression efficiency, only focusing on motion information between frames. Also, the pre-trained PWC-Net is trained with a smooth motion constraint. However, the motion information from the fine-tuned PWC-Net contains both smooth and textured motion fields since it extracts motion information in a rate-distortion sense. Therefore, the optical flow with the texture parts that are generated by the fine-tuned PWC-Net is more suitable for video compression than the optical flow with the smooth parts that are generated by the pre-trained PWC-Net.

Subjective Visual Comparisons: Fig. 9 shows some cropped regions of decoded frames of *HoneyBee* and *YachtRide* sequences by H.264, H.265 and our method for visual comparisons. Our method yields decoded frames with higher contrast and less artifact than H.264 and H.265. The decoded results by H.264 and H.265 show that the wing and leg of the honey bee are poorly reconstructed, but our DeepPVCnet reconstructs those with a higher contrast and less artifacts. Also, similar results in a low bitrate range are observed for *BQTerrace* and *Cactus* sequences as shown in Fig. 10. Similarly, Fig. 11 shows some cropped regions of decoded frames of *Bosphorus* and *Kimono* sequences by H.264, H.265 and our method for visual comparisons. As shown in Fig. 11, while the H.264 and H.265 produce the the decoded regions with blocking artifacts in a low bitrate range, our method yields the decoded region of higher fidelity without such artifacts.

V. CONCLUSION

We propose an end-to-end deep predictive video compression network, called DeepPVCnet, based on multi-frame hypothesis with a multi-scale structure and a temporal-context-adaptive entropy model. Our DeepPVCnet incorporates a mode-selective framework with uni- and bi-directional predictive codings in a rate-distortion optimization sense by jointly compressing optical flows and residual data that are generated from the multi-scale structure via the FT layers in an encoder side. In addition, our DeepPVCnet with the temporal-context-adaptive entropy model has a much faster decoding speed because it can be performed in parallel unlike the recent video compression methods [11], [13] using the autoregressive-based entropy coding model. Based on these advanced components in a combination, the DeepPVCnet shows better compression performance than the existing video standard compression codecs (AVC/H.264 and HEVC/H.265) and recent SOTA methods in terms of MS-SSIM. In our future work, our DeepPVCnet is extended to learn a fully automatic selection of the best prediction modes during training.

APPENDIX A
THE COMMANDS OF CONVENTIONAL VIDEO CODECS

For the implementation of H.264 [45] and H.265 [38] with several options, we used *ffmpeg* and x265 [3] to compress the sequences, respectively, as follows:

- P_veryfast (P-frame with veryfast)

```
> ffmpeg -s:v HxW -framerate FR
  -i input.yuv -vcodec libx264 -crf QP
  -bf 0 -b_strategy 0 -sc_threshold 0
  -preset veryfast -tune zerolatency
  -g G -keyint_min G -pix_fmt yuv420p
  output.mp4
> x265 --profile main --level 5
  -p veryfast --tune zerolatency
  --crf QP --keyint G --min-keyint G
  --input-res HxW --fps FR
  --input input.yuv -o output.mp4
```

- P_medium (P-frame with medium)

```
> ffmpeg -s:v HxW -framerate FR
  -i input.yuv -vcodec libx264 -crf QP
  -bf 0 -b_strategy 0 -sc_threshold 0
  -preset medium -g G -keyint_min G
  -pix_fmt yuv420p output.mp4
> x265 --profile main --level 5
  -p medium --bframes 0 --crf QP
  --keyint G --min-keyint G
  --input-res HxW --fps FR
  --input input.yuv -o output.mp4
```

- B_veryfast (B-frame with veryfast)

```
> ffmpeg -s:v HxW -framerate FR
  -i input.yuv -vcodec libx264 -crf QP
  -preset veryfast -g G -keyint_min G
  -pix_fmt yuv420p output.mp4
> x265 --profile main --level 5
  -p veryfast --crf QP --keyint G
  --min-keyint G --input-res HxW
  --fps FR --input input.yuv
  -o output.mp4
```

- B_medium (B-frame with medium, default)

```
> ffmpeg -s:v HxW -framerate FR
  -i input.yuv -vcodec libx264 -crf QP
  -preset medium -g G -keyint_min G
  -pix_fmt yuv420p output.mp4
> x265 --profile main --level 5
  -p medium --crf QP --keyint G
  --min-keyint G --input-res HxW
  --fps FR --input input.yuv
  -o output.mp4
```

- SSIM (B-frame with medium, ssim)

```
> ffmpeg -s:v HxW -framerate FR
  -i input.yuv -vcodec libx264 -crf QP
  -preset medium -tune ssim -g G
  -keyint_min G -pix_fmt yuv420p
  output.mp4
```

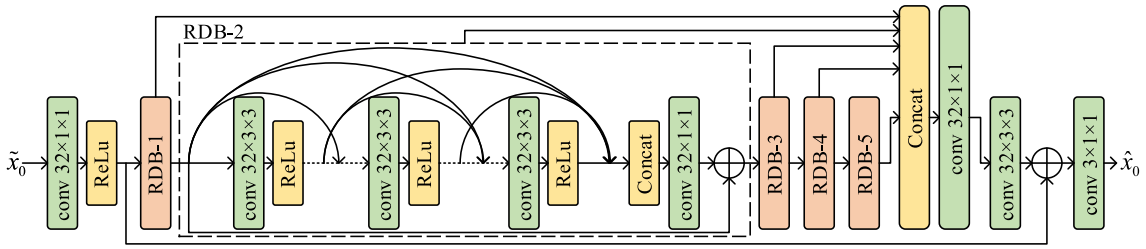

FIGURE 12. The Enhancement Net for our DeepPVCnet.

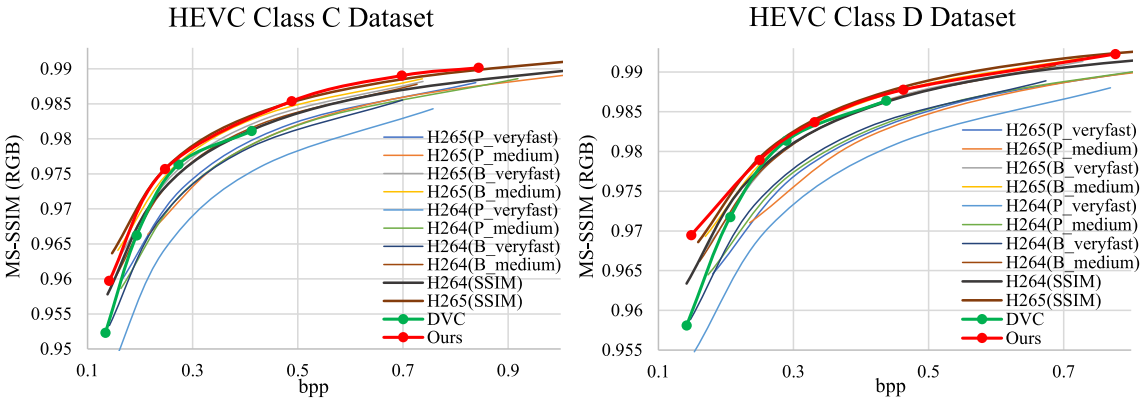

FIGURE 13. MS-SSIM performance comparison of our DeepPVCnet, H.264 [45] and H.265 [38], and DVC [26] for the HEVC Class C and D dataset.

```
> x265 --profile main --level 5
  -p medium --tune ssim --crf QP
  --keyint G --min-keyint G
  --input-res HxW --fps FR
  --input input.yuv -o output.mp4
```

where (H, W), FR, QP and G denote the spatial resolutions, the framerate, the quantization parameter and the GOP size, respectively.

APPENDIX B
THE IMPLEMENTATION OF OUR PROPOSED ENTROPY CODING MODEL

For more details of the implementation of our proposed entropy coding model, we follow the same concept and notations in the CNN-based image compression methods [6], [21]. In the main paper, we provided the training loss L as the rate-distortion optimization problem for video compression. Since the quantization of the latent representation is discrete, we substitute additive uniform noise for the quantization process during training. Then the approximated latent representations $\tilde{y}_0$ and $\tilde{z}_0$ are used instead of the quantized latent representations $\hat{y}_0$ and $\hat{z}_0$, respectively, in the training loss L as follows:

$$L \approx \mathbb{E}_{x_0\sim p_{x_0}}\mathbb{E}_{\tilde{y}_0,\tilde{z}_0\sim q}[-\log_2 p_{\tilde{y}_0|(\tilde{z}_0,X^R)}(\tilde{y}_0|(\tilde{z}_0,X^R)) - \log_2 p_{\tilde{z}_0}(\tilde{z}_0) + \lambda\cdot d(x_0,\hat{x}_0)], \quad (7)$$

where x_0, $\hat{x}_0$ and X^R denote the current frame to be encoded, the reconstructed frame and the reference frames for x_0, respectively. The joint factorized posterior with the additive uniform noise for the quantization process as in [6], [21] can be expressed as follows:

$$q(\tilde{y}_0,\tilde{z}_0|x_0,\phi_g,\phi_h) = \prod_i \mathcal{U}(\tilde{y}_{0,i}|\tilde{y}_{0,i}-\tfrac{1}{2},\tilde{y}_{0,i}+\tfrac{1}{2}) \cdot \prod_i \mathcal{U}(\tilde{z}_{0,i}|\tilde{z}_{0,i}-\tfrac{1}{2},\tilde{z}_{0,i}+\tfrac{1}{2}) \quad \text{with } y=g_a(x_0;\phi_g), z=h_a(y_0;\phi_h), \quad (8)$$

where $\mathcal{U}$, ϕ_g and ϕ_h denote a uniform distribution, the parameters of g_a and h_a, respectively. Our proposed entropy coding model approximates the required bits for $\hat{y}_0$ and $\hat{z}_0$ as in Eq. 7. The entropy coding model for $\hat{y}_0$ is based on Gaussian model with mean μ_i and standard deviation σ_i. Our proposed Context-Net C and the hyper encoder-decoder network pair (h_a, h_s) with the multiple reference frames X^R estimate the values of μ_i and σ_i. The Context-Net C generates the temporal context information c_t from X^R and the hyper encoder-decoder network generates the context information c_c from y_0. Then the Context-Net concatenates c_t and c_c to estimate the values of μ_i and σ_i. The expression for this process is as follows:

$$p_{\tilde{y}_0|\tilde{z}_0}(\tilde{y}_0|\tilde{z}_0,X^R,\theta_c,\theta_h) = \prod_i (\mathcal{N}(\mu_i,\sigma_i^2) * \mathcal{U}(-\tfrac{1}{2},\tfrac{1}{2}))(\tilde{y}_{0,i}) \quad \text{with } \mu_i,\sigma_i = C(X^R,c_c;\theta_c), \; c_c = h_s(\tilde{z}_0;\theta_h), \quad (9)$$

where θ_c and θ_h denote the parameters of the Context-Net C and the hyper decoder network h_s. Note that our proposed

entropy coding model can estimate *mu* and *sigma* in parallel during decoding process since the entropy coding model with the multiple reference frames is not autoregressive. We utilized the same entropy coding model for $\hat{z}_0$ which follows a zero-mean Gaussian model with standard deviation σ as in [6]. Since $\hat{z}_0$ has little effect on the total bit-rate of the current frame coding, we use a simpler entropy coding model for $\hat{z}_0$ than the entropy coding model of $\hat{y}_0$ as follows:

$$p_{\tilde{z}_0}(\tilde{z}_0) = \prod_i (\mathcal{N}(0, \sigma_i^2) * \mathcal{U}(-\tfrac{1}{2}, \tfrac{1}{2}))(\tilde{z}_{0,i}). \tag{10}$$

APPENDIX C
THE ARCHITECTURE OF ENHANCEMENT NET

In the main paper, we described the overall structure of our DeepPVCnet that consists of an encoder-decoder network pair (g_a, g_s) with the feature transformation layer, a hyper encoder-decoder network pair (h_a, h_s), the pre-trained PWC-Net [39], a Context-Net and an Enhancement Net. The Enhancement Net is incorporated into the decoder side of our DeepPVCnet to enhance the image quality of the reconstructed frame $\tilde{x}_0$. Fig. 12 shows the details of the Enhancement Net that consists of the residual dense network (RDN) [50]. As depicted in Fig. 12, the Enhancement Net consists of five residual dense blocks (RDB) with three convolution filters per each block.

APPENDIX D
THE EXPERIMENTAL RESULTS FOR HEVC CLASS C AND D IN MS-SSIM

In the main paper, we showed the results of the rate-distortion (R-D) curves for our DeepPVCnet, H.264, H.265, Wu's method [46], DVC [26] and Habibian's method [13] with the UVG [2] and HEVC datasets [38] (Class B and E) that are consist of high-resolution sequences. Additionally, Fig. 13 shows the results of the R-D curves in terms of MS-SSIM for the HEVC datasets (Class C and D) that are low-resolution sequences. Our DeepPVCnet outperforms H.264, H.265 and DVC for most bitrate ranges in terms of MS-SSIM. Note that the experimental results for the HEVC datasets are provided only by DVC [26] among the recent deep video compression methods.

REFERENCES

[1] *UGC Dataset*. Accessed: Jun. 3, 2020. [Online]. Available: https://media.withyoutube.com/
[2] *Ultra Video Group Test Sequences*. Accessed: Jun. 3, 2020. [Online]. Available: http://ultravideo.cs.tut.fi/
[3] *x265 HEVC Encoder/h.265 Video Codec*. Accessed: Jun. 3, 2020. [Online]. Available: http://x265.org/
[4] E. Agustsson, D. Minnen, N. Johnston, J. Balle, S. J. Hwang, and G. Toderici, "Scale-space flow for end-to-end optimized video compression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, May 2020, pp. 8503–8512.
[5] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," 2016, *arXiv:1611.01704*. [Online]. Available: http://arxiv.org/abs/1611.01704
[6] J. Ballé, D. Minnen, S. Singh, S. Jin Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," 2018, *arXiv:1802.01436*. [Online]. Available: http://arxiv.org/abs/1802.01436
[7] F. Bellard. (2015). *BPG Image Format*. [Online]. Available: http://bellard.org/bpg/
[8] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *Proc. 13th Meeting Video Coding Experts Group (VCEG)*, Austin, TX, USA, Apr. 2001, pp. 1–4.
[9] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learning image and video compression through spatial-temporal energy compaction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 10071–10080.
[10] W. Cui, T. Zhang, S. Zhang, F. Jiang, W. Zuo, and D. Zhao, "Convolutional neural networks based intra prediction for HEVC," 2018, *arXiv:1808.05734*. [Online]. Available: http://arxiv.org/abs/1808.05734
[11] A. Djelouah, J. Campos, S. Schaub-Meyer, and C. Schroers, "Neural inter-frame compression for video coding," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 6421–6429.
[12] 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.* Springer, 2014, pp. 184–199.
[13] A. Habibian, T. V. Rozendaal, J. M. Tomczak, and T. S. Cohen, "Video compression with rate-distortion autoencoders," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 7033–7042.
[14] C. Jia, S. Wang, X. Zhang, S. Wang, J. Liu, S. Pu, and S. Ma, "Content-aware convolutional neural network for in-loop filtering in high efficiency video coding," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3343–3356, Jan. 2019.
[15] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz, "Super SloMo: High quality estimation of multiple intermediate frames for video interpolation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9000–9008.
[16] N. Johnston, D. Vincent, D. Minnen, M. Covell, S. Singh, T. Chinen, S. J. Hwang, J. Shor, and G. Toderici, "Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4385–4393.
[17] 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. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 26–30.
[18] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1646–1654.
[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[20] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4681–4690.
[21] J. Lee, S. Cho, and S.-K. Beack, "Context-adaptive entropy model for end-to-end optimized image compression," 2018, *arXiv:1809.10452*. [Online]. Available: http://arxiv.org/abs/1809.10452
[22] C. Li, L. Song, R. Xie, and W. Zhang, "CNN based post-processing to improve HEVC," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 4577–4580.
[23] M. Li, W. Zuo, S. Gu, D. Zhao, and D. Zhang, "Learning convolutional networks for content-weighted image compression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3214–3223.
[24] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 136–144.
[25] J. Lin, D. Liu, H. Li, and F. Wu, "M-LVC: Multiple frames prediction for learned video compression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3546–3554.
[26] G. Lu, W. Ouyang, D. Xu, X. Zhang, C. Cai, and Z. Gao, "DVC: An end-to-end deep video compression framework," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11006–11015.
[27] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. V. Gool, "Conditional probability models for deep image compression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4394–4402.
[28] S. Meyer, A. Djelouah, B. McWilliams, A. Sorkine-Hornung, M. Gross, and C. Schroers, "PhaseNet for video frame interpolation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 498–507.

[29] D. Mukherjee, J. Bankoski, A. Grange, J. Han, J. Koleszar, P. Wilkins, Y. Xu, and R. Bultje, "The latest open-source video codec VP9–An overview and preliminary results," in *Proc. Picture Coding Symp. (PCS)*, Dec. 2013, pp. 390–393.
[30] S. Niklaus and F. Liu, "Context-aware synthesis for video frame interpolation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1701–1710.
[31] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive convolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 670–679.
[32] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 261–270.
[33] 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 (IVMSP)*, Jul. 2016, pp. 1–5.
[34] A. Ranjan and M. J. Black, "Optical flow estimation using a spatial pyramid network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4161–4170.
[35] O. Rippel and L. Bourdev, "Real-time adaptive image compression," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, vol. 70, pp. 2922–2930.
[36] O. Rippel, S. Nair, C. Lew, S. Branson, A. G. Anderson, and L. Bourdev, "Learned video compression," 2018, *arXiv:1811.06981*. [Online]. Available: http://arxiv.org/abs/1811.06981
[37] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Process. Mag.*, vol. 18, no. 5, pp. 36–58, May 2001.
[38] 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.
[39] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8934–8943.
[40] L. Theis, W. Shi, A. Cunningham, and F. Huszár, "Lossy image compression with compressive autoencoders," 2017, *arXiv:1703.00395*. [Online]. Available: http://arxiv.org/abs/1703.00395
[41] G. Toderici, S. M. O'Malley, S. Jin Hwang, D. Vincent, D. Minnen, S. Baluja, M. Covell, and R. Sukthankar, "Variable rate image compression with recurrent neural networks," 2015, *arXiv:1511.06085*. [Online]. Available: http://arxiv.org/abs/1511.06085
[42] G. Toderici, D. Vincent, N. Johnston, S. J. Hwang, D. Minnen, J. Shor, and M. Covell, "Full resolution image compression with recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5306–5314.
[43] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. 18–19, Feb. 1992.
[44] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, vol. 2. Nov. 2003, pp. 1398–1402.
[45] 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.
[46] C.-Y. Wu, N. Singhal, and P. Krahenbuhl, "Video compression through image interpolation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 416–431.
[47] R. Yang, F. Mentzer, L. Van Gool, and R. Timofte, "Learning for video compression with hierarchical quality and recurrent enhancement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6628–6637.
[48] R. Yang, M. Xu, Z. Wang, and T. Li, "Multi-frame quality enhancement for compressed video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6664–6673.
[49] 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. 3827–3841, Aug. 2018.
[50] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2472–2481.
[51] Z. Zhao, S. Wang, S. Wang, X. Zhang, S. Ma, and J. Yang, "Enhanced bi-prediction with convolutional neural network for high efficiency video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 3291–3301, Nov. 2018.

WOONSUNG PARK received the B.S. and M.S. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree with the Department of Electrical Engineering. His current research interests include deep learning-based frame interpolation and image/video compression.

MUNCHURL KIM (Senior Member, IEEE) received the B.E. degree in electronics from Kyungpook National University, Daegu, South Korea, in 1989, and the M.E. and Ph.D. degrees in electrical and computer engineering from the University of Florida, Gainesville, FL, USA, in 1992 and 1996, respectively. He joined the Electronics and Telecommunications Research Institute, Daejeon, South Korea, as a Senior Research Staff Member, where he led the Realistic Broadcasting Media Research Team. In 2001, he joined as an Assistant Professor with the School of Engineering, Information and Communications, Daejeon. Since 2009, he has been with the School of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, where he is currently a Full Professor. He has also been involved in scalable video coding and high-efficiency video coding in JCT-VC standardization activities of ITUT VCEG and ISO/IEC MPEG. His current research interests include high-performance video coding, perceptual video coding, visual quality assessments on 3D/UHD video, visual information processing, machine learning, and pattern recognition.