Neural Video Compression with Context Modulation

Chuanbo Tang, Zhuoyuan Li, Yifan Bian, Li Li, Dong Liu

MOE Key Laboratory of Brain-Inspired Intelligent Perception and Cognition
University of Science and Technology of China, Hefei 230027, China
{cbtang,zhuoyuanli}@mail.ustc.edu.cn, togelbian@gmail.com, {lil1,dongeliu}@ustc.edu.cn

6. Appendices

This document provides supplementary material to our proposed neural video codec (NVC), i.e., DCMVC.

6.1. Network Structure

Our DCMVC is built on DCVC-DC [27], and proposes context modulation to generate high-quality temporal context exploiting the reference information in both pixel and feature domain. Our proposed context modulation consists of two modules: flow orientation and context compensation. The workflow of flow orientation has already been demonstrated in detail. Here, we describe the details of context compensation in network structure, which can be divided into two parts: feature extraction and feature fusion.

6.1.1. Feature Extraction

First, the oriented temporal context $\check{C}_t^0$ and propagated temporal context C_t^0 are input to the mutual extractors sharing the same network parameters to extract the shallow features, respectively. As shown in Fig. 9, we regard the oriented temporal context $\check{C}_t^0$ as the input to illustrate the network structure of mutual extractor. For convolution layer, the $(K, Cin, Cout, S)$ indicates the kernel size, input channel number, output channel number, and stride, respectively. After the first convolution layer and activation layer, the extracted features are processed using two types of basic blocks: ResidualBlock and DepthConvBlock.

The structure of ResidualBlock is shown in Fig. 10 (a), where the input is added to the features extracted from the middle layers. As illustrated in Fig. 10 (b), DepthConvBlock which contains depthwise separable convolution and pointwise convolution is used for reducing the computation cost. After obtaining the shallow features from the mutual extractor, the subsequent private global and local extrators learn the global and local features of each temporal context individually. The structure of global extractor is shown in Fig. 11, except the first depthwise convolution layer is with 3×3 kernel size, other convolutional layers are designed with 1×1 kernel size for reducing the complexity. We wish the local branch can preserve as much detailed information of two temporal contexts as possible, so we adopt invertible neural networks with affine decoupling layers as shown in Fig. 12 (a). The detailed network structure of Inverted Blocks utilized in the local extractor is shown in Fig. 12 (b).

Corresponding author: Dong liu

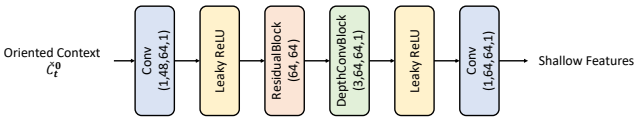

Figure 9. The network structure of mutual extractor.

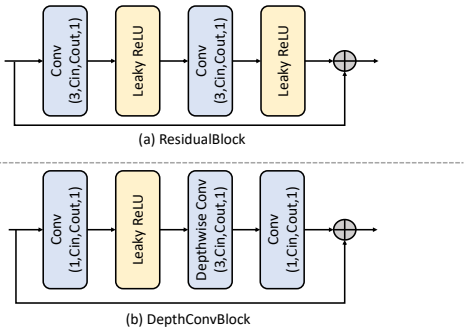

Figure 10. (a) The network structure of ResidualBlock. (b) The network structure of DepthConvBlock.

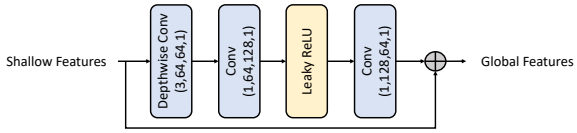

Figure 11. The network structure of global extractor. The global fusion shares the same network architecture with global extractor.

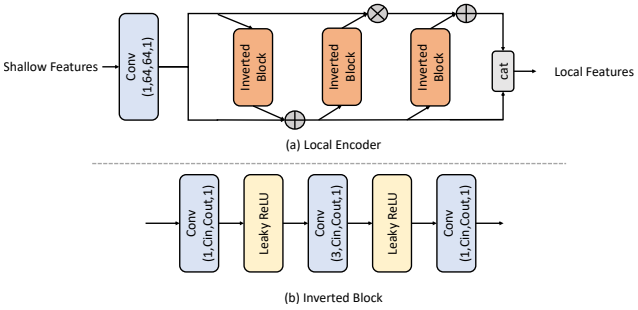

Figure 12. (a) The network structure of local extractor, and the local fusion shares the same network architecture with local extractor. (b) The network structure of Inverted Block.

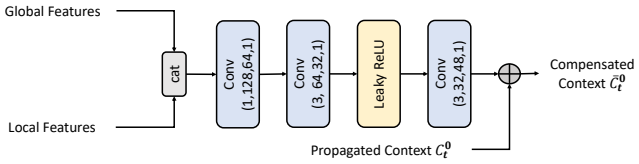

Figure 13. The network structure of mutual fusion.

Table 5. BD-Rate (%) of using different α to control the weight of decoupling loss.

α	0	0.1	0.2	0.4
BD-Rate (%)	0.0	-0.2	-1.0	-0.5

6.1.2. Feature Fusion

To maintain the performance by leveraging the consistency assumption, the networks of extractor and fusion adopt the same architecture. The network structures of global and local fusion are also shown in Fig. 11 and Fig. 12 (a). After obtaining the fused global feature and local feature, the two features are concatenated to input the mutual fusion as shown in Fig. 13. The first convolution layer with 1×1 kernel size is utilized for reducing the channel number, and the subsequent two convolution layers with 3×3 kernel size are utilized to extract the features to compensate the propagated temporal context C_t^0 for generating the final compensated temporal context $\overline{C}_t^0$.

6.2. Decoupling Loss

In this paper, we design the decoupling loss to facilitate the synergy mechanism in context compensation during the training. When adding the decoupling loss into the rate-distortion loss function, the α is set for controlling the weight of decoupling loss. We train models with different α, and evaluate the average rate-distortion (RD) performance on HEVC datasets with the intra-period set of 32 shown in Table 5. The model M_d in Table 3 is the anchor with α set as 0 in Table 5. From this table, we can see that when α is set as 0.2, the model achieves the highest compression ratio, so we choose 0.2 as our final setting for α.

6.3. Rate-Distortion Curves

In this document, we demonstrate the RD curves of all testing datasets in terms of RGB-PSNR, and the intra-period is set as 32 and -1, respectively. We evaluate our scheme against two categories of codecs. For the traditional codec, we choose H.266/VVC [8] as our benchmark. For NVCs, we choose six benchmarks: DCVC [25], DCVC-TCM [47], DCVC-HEM [26], DCVC-DC [27], SDD [48], and DCVC-FM [28].

As shown in Fig. 14, we illustrate the RD curves of USTC-TD [33] dataset under the intra-period setting of

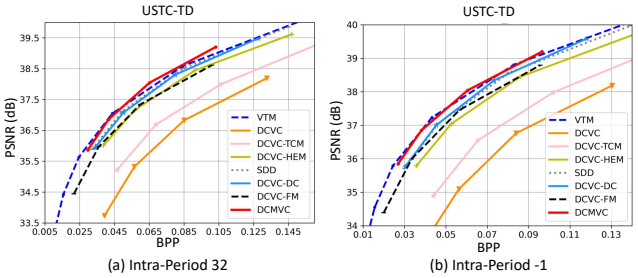

Figure 14. Rate and distortion curve for USTC-TD dataset, and the comparison is in RGB colorspace measured with PSNR. (a) The intra-period is set as 32. (b) The intra-period is set as -1.

32 and -1. All the other NVCs that we compared show a notable gap in compression performance compared to H.266/VVC under both intra-period settings, while our DCMVC achieves compression performance most comparable to H.266/VVC, demonstrating its superior effectiveness. The performance loss of NVCs in the USTC-TD dataset may come from the presence of complex motion features, such as high-speed moving objects and object occlusions, which lead to poor temporal prediction. In the future, we will address this limitation in three ways: first, we will design a learnable warp operation for flexible temporal alignment. Second, we will learn multiple predefined motion patterns as priors for more efficient temporal context modeling. Third, we will consider adding long-term temporal dependencies to our temporal context modeling for utilizing multi-frame information.

Moreover, we also show the RD curves of UVG [42], MCl-JCV [55], HEVC B, C, D, and E [7] datasets under intra-period setting of 32 in Fig. 15 and Fig. 16. From these figures, we can see our DCMVC achieves state-of-the-art (SOTA) compression ratio in all the datasets. As shown in Fig. 17 and Fig. 18, the RD curves of aforementioned datasets under intra-period setting of -1 are illustrated. We can see our DCMVC outperforms other schemes significantly in terms of compression ratio, which verifies the effectiveness of our scheme in long prediction chains.

6.4. Visual Comparison

We provide four visual comparisons across difference sequences to show the advantage of our DCMVC, which are shown in Fig 20 and Fig 19. From these figures, we can see that our DCMVC enables better reconstruction of both structural and texture information without introducing additional bitrate cost, when compared to the previous SOTA NVCs, DCVC-DC and DCVC-FM.

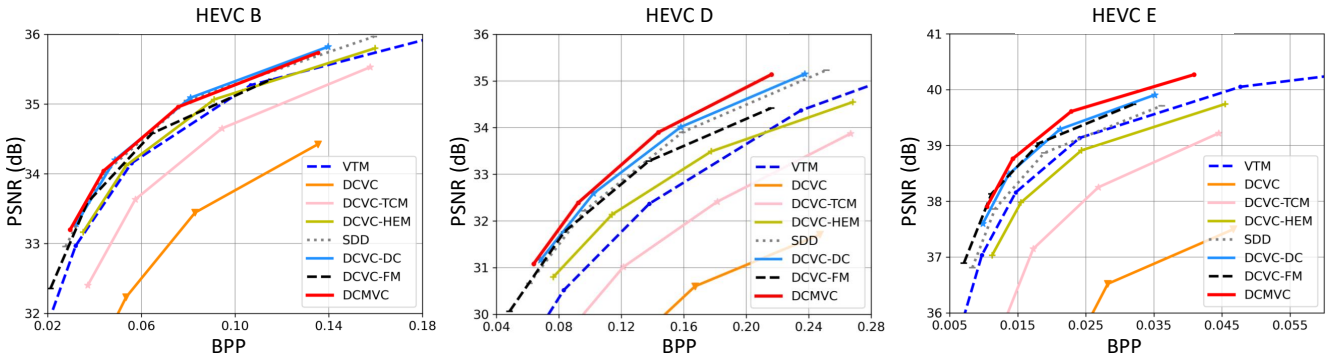

Figure 15. Rate and distortion curve for HEVC Class B, D, and E datasets. The comparisons are in RGB colorspace measured with PSNR, and the intra-period is set as 32.

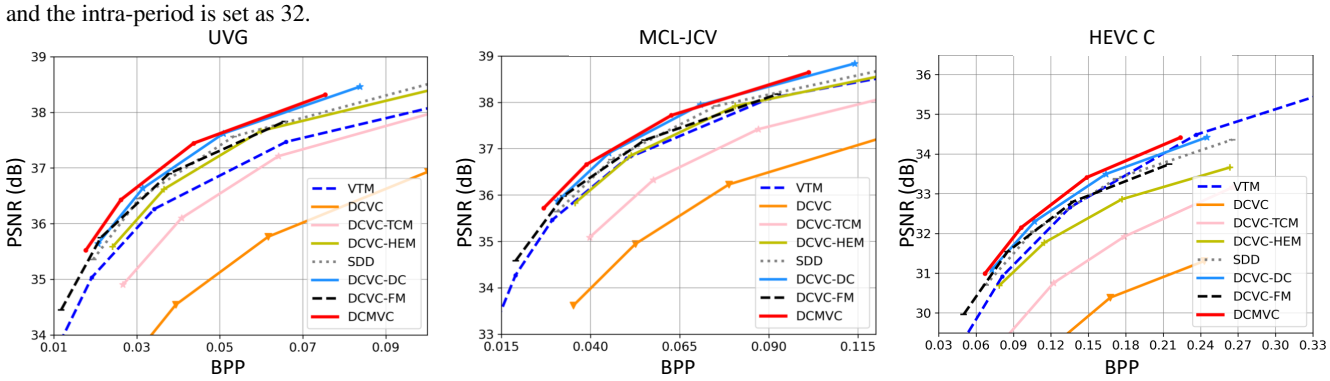

Figure 16. Rate and distortion curve for UVG, MCL-JCV, and HEVC Class C datasets. The comparisons are in RGB colorspace measured with PSNR, and the intra-period is set as 32.

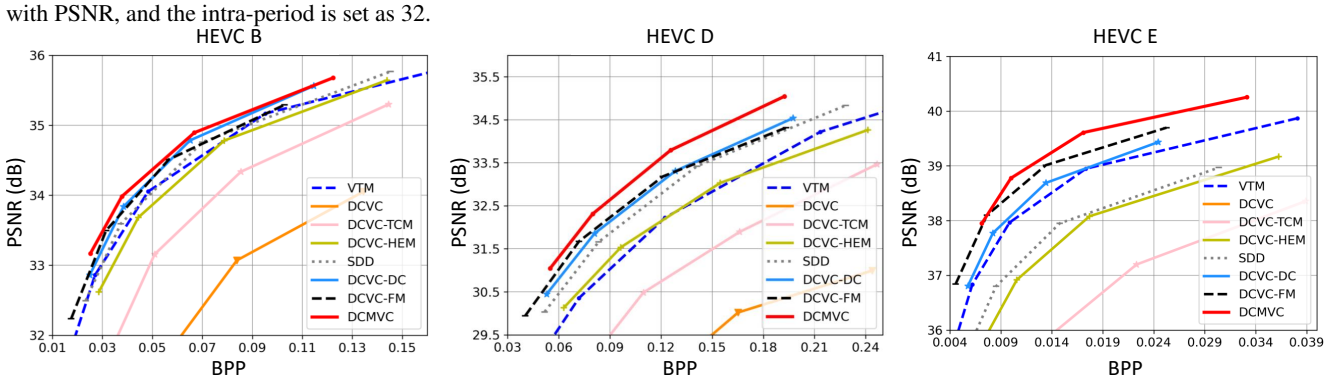

Figure 17. Rate and distortion curve for HEVC Class B, D, and E datasets. The comparisons are in RGB colorspace measured with PSNR, and the intra-period is set as -1.

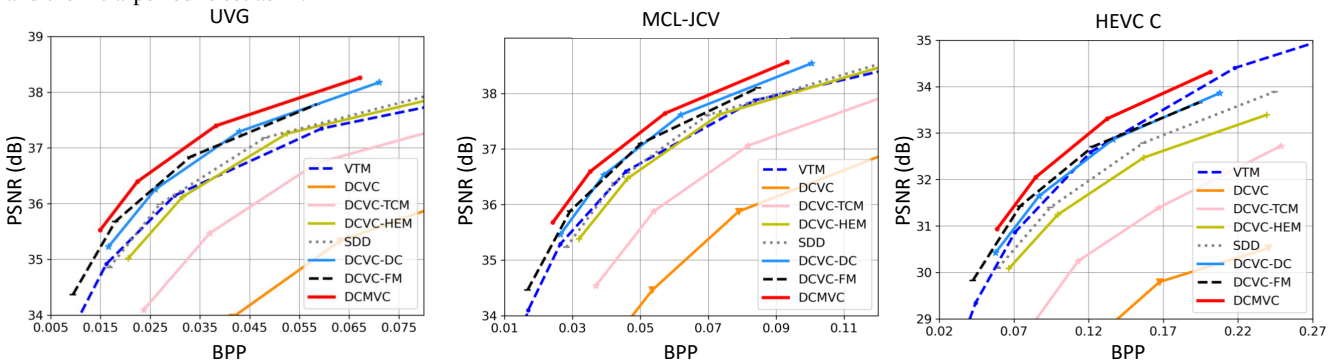

Figure 18. Rate and distortion curve for UVG, MCL-JCV, and HEVC Class C datasets. The comparisons are in RGB colorspace measured with PSNR, and the intra-period is set as -1.

Figure 19. Visual comparisions for Jockey and BasketballDrill sequences.

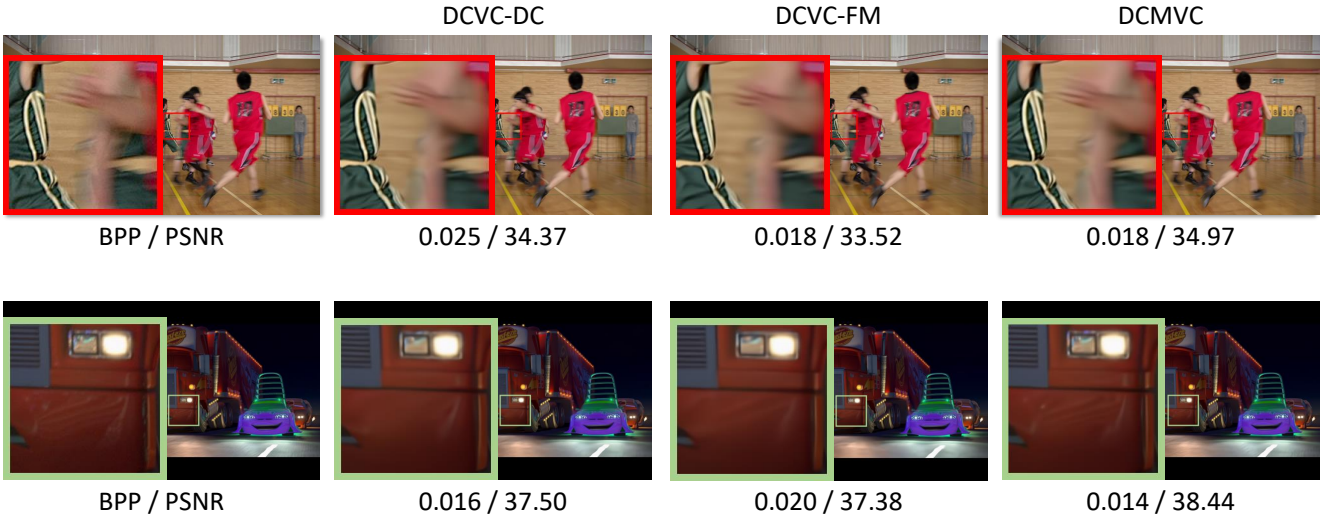

Figure 20. Visual comparisions for BasketballDrive and videoSRC24 sequences.