Neural Video Compression with Diverse Contexts

Jiahao Li, Bin Li, Yan Lu
Microsoft Research Asia
{li.jiahao, libin, yanlu}@microsoft.com

Abstract

For any video codecs, the coding efficiency highly relies on whether the current signal to be encoded can find the relevant contexts from the previous reconstructed signals. Traditional codec has verified more contexts bring substantial coding gain, but in a time-consuming manner. However, for the emerging neural video codec (NVC), its contexts are still limited, leading to low compression ratio. To boost NVC, this paper proposes increasing the context diversity in both temporal and spatial dimensions. First, we guide the model to learn hierarchical quality patterns across frames, which enriches long-term and yet high-quality temporal contexts. Furthermore, to tap the potential of optical flow-based coding framework, we introduce a group-based offset diversity where the cross-group interaction is proposed for better context mining. In addition, this paper also adopts a quadtree-based partition to increase spatial context diversity when encoding the latent representation in parallel. Experiments show that our codec obtains 23.5% bitrate saving over previous SOTA NVC. Better yet, our codec has surpassed the under-developing next generation traditional codec/ECM in both RGB and YUV420 colorspaces, in terms of PSNR. The codes are at https://github.com/microsoft/DCVC.

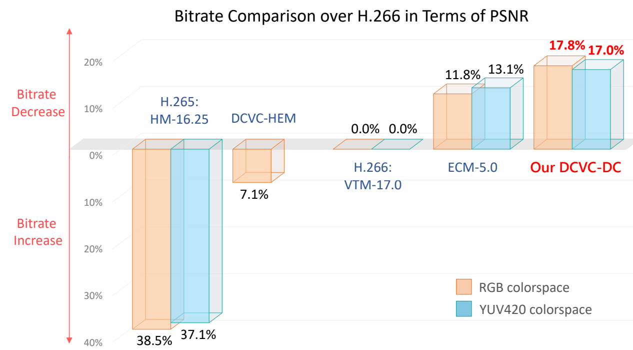

Figure 1. Average results on UVG, MCL-JCV, and HEVC datasets. All traditional codecs use their best compression ratio configuration. DCVC-HEM [29] is the previous SOTA NVC and only has released the model for RGB colorspace.

1. Introduction

The philosophy of video codec is that, for the current signal to be encoded, the codec will find the relevant contexts (e.g., various predictions as the contexts) from previous reconstructed signals to reduce the spatial-temporal redundancy. The more relevant contexts are, the higher bitrate saving is achieved.

If looking back the development of traditional codecs (from H.261 [17] in 1988 to H.266 [7] in 2020), we find that the coding gain mainly comes from the continuously expanded coding modes, where each mode uses a specific manner to extract and utilize context. For example, the numbers of intra prediction directions [42] in H.264, H.265, H.266 are 9, 35, and 65, respectively. So many modes can extract diverse contexts to reduce redundancy, but also bring huge complexity as rate distortion optimization (RDO) is used to search the best mode. For encoding a 1080p frame, the under-developing ECM (the prototype of next generation traditional codec) needs up to half an hour [49]. Although some DL-based methods [24,51,52] proposed accelerating traditional codecs, the complexity is still very high.

By contrast, neural video codec (NVC) changes the extraction and utilization of context from hand-crafted design to automatic-learned manner. Mainstream frameworks of NVC can be classified into residual coding-based [1, 13, 31, 32, 34, 36, 47, 59, 61] and condition coding-based [21, 27–29, 33, 38, 50]. The residual coding explicitly uses the predicted frame as the context, and the context utilization is restricted to use subtraction for redundancy removal. By comparison, conditional coding implicitly learns feature domain contexts. The high dimension contexts can carry richer information to facilitate encoding, decoding, as well as entropy modelling.

However, for most NVCs, the manners of context extraction and utilization are still limited, e.g., only using optical flow to explore temporal correlation. This makes NVC easily suffer from the uncertainty [12, 16, 37] in parameters or fall into local optimum [25]. One solution is adding traditional codec-like coding modes into NVC [25]. But it brings large computational complexity as RDO is used. So the question comes up: how to better learn and use the contexts while yielding low computational cost?

To this end, based on DCVC (deep contextual video compression) [28] framework and its following work DCVC-HEM [29], we propose a new model DCVC-DC which efficiently utilizes the Diverse Contexts to further boost compression ratio. At first, we guide DCVC-DC to learn hierarchical quality pattern across frames. With this guidance during the training, the long-term and yet high-quality contexts which are vital for the reconstruction of the following frames are implicitly learned during the feature propagation. This helps further exploit the long-range temporal correlation in video and effectively alleviate the quality degradation problem existed in most NVCs. In addition, we adopt the offset diversity [8] to strengthen the optical flow-based codec, where multiple offsets can reduce the warping errors for complex or large motions. In particular, inspired by the weighted prediction in traditional codec, the offsets are divided into groups and the cross-group fusion is proposed to improve the temporal context mining.

Besides from temporal dimension, this paper also proposes increasing the spatial context diversity when encoding the latent representation. Based on recent checkerboard model [19] and dual spatial model [29, 56], we design a quadtree-based partition to improve the distribution estimation. When compared with [19, 29], the types of correlation modelling are more diverse hence the model has a larger chance to find more relevant context.

It is noted that all our designs are parallel-efficient. To further reduce the computational cost, we also adopt depth-wise separable convolution [10], and assign unequal channel numbers for features with different resolutions. Experiments show that our DCVC-DC achieves much higher efficiency over previous SOTA NVC and pushes the compression ratio to a new height. When compared with DCVC-HEM [29], 23.5 % bitrate saving is achieved while MACs (multiply–accumulate operations) are reduced by 19.4%. Better yet, besides H.266-VTM 17.0, our codec also already outperforms ECM-5.0 (its best compression ratio configuration for low delay coding is used) in both RGB and YUV420 colorspaces, as shown in Fig. 1. To the best of our knowledge, this is the first NVC which can achieve such accomplishment. In summary, our contributions are:

- We propose efficiently increasing context diversity to boost NVC. Diverse contexts are complementary to each other and have larger chance to provide good reference for reducing redundancy.
- From temporal dimension, we guide model to extract high-quality contexts to alleviate the quality degradation problem. In addition, the group-based offset diversity is designed for better temporal context mining.
- From spatial dimension, we adopt a quadtree-based partition for latent representation. This provides diverse spatial contexts for better entropy coding.
- Our DCVC-DC obtains 23.5% bitrate saving over the previous SOTA NVC. In particular, our DCVC-DC has surpassed the best traditional codec ECM in both RGB and YUV420 colorspaces, which is an important milestone in the development of NVC.

2. Related Work

2.1. Neural Image Compression

Most neural image codecs are based on hyperprior [4] where some bits are first used to provide basic contexts for entropy coding. Then, the auto-regressive prior [40] proposes using neighbour contexts to capture spatial correlation. Recently works [18, 26, 44, 45] propose extracting global or long-range contexts to further boost performance. These show more diverse contexts bring substantial coding gain for neural image codec.

2.2. Neural Video Compression

Recent years also have witnessed the prosperity of NVC. The pioneering DVC [34] follows traditional codec. It uses optical flow network to generate prediction frame, then its residual with the current frame is coded. Many subsequent works also adopt this residual coding-based framework and refine the modules therein. For example, [31, 43, 47] proposed motion prediction to further reduce redundancy. Optical flow estimation in scale-space [1] was designed to handle complex motion. Yang *et al.* [61] utilized recurrent auto-encoder to improve coding efficiency.

Residual coding explicitly generates predicted frame in pixel domain as the context and only uses the subtraction to remove redundancy. By comparison, conditional coding has stronger extensibility. The definition, learning, and usage manner of condition can be flexibly designed. In [33, 38], temporally conditional entropy models were designed. [27] uses conditional coding to encode the foreground contents. Li *et al.* proposed DCVC [28] to learn feature domain contexts to increase context capacity. Then DCVC-TCM [50] adopts feature propagation to boost performance. Recently, DCVC-HEM [29] designs the hybrid entropy model utilizing both spatial and temporal contexts.

However, the coding modes in most NVCs are still limited when compared with traditional codec. For example, traditional codec adopts translational/affine motion models, geometric partition, bi-prediction, and so on modes to extract diverse temporal contexts [7]. By contrast, existing NVCs usually only relies on single optical flow, which is easily influenced by epistemic uncertainty [12, 16, 37] in model parameters. The recent work [25] also shows such NVCs easily fall into local optimum when coding mode is

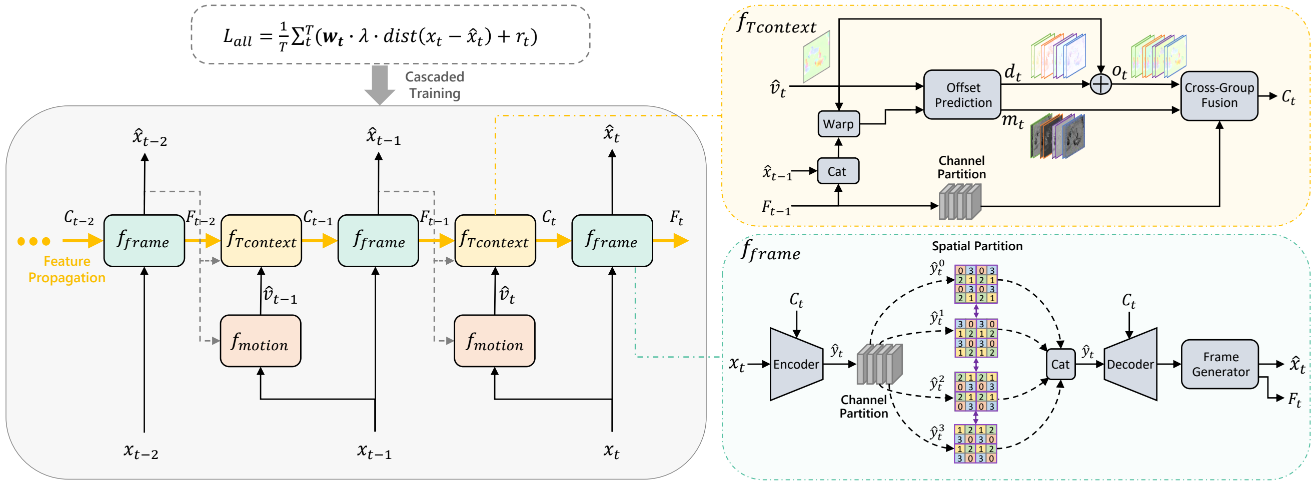

Figure 2. Framework overview of our DCVC-DC. x_t and $\hat{x}_t$ are the input and reconstructed frames. C_t is the learned temporal context as the condition for coding x_t. F_t is the propagated but unprocessed feature used for next frame. In the loss term, r_t means the bit cost for coding whole frame. $dist(\cdot)$ is distortion function. λ and w_t are the global and frame-level weights, respectively. The number in the spatial partition for $\hat{y}_t$ represent the coding order index.

limited. So [25] designed many additional modes like traditional codec, and RDO is used to search the best mode. Such method inevitably brings large computational cost. By comparison, our model has no additional inference cost when exploiting high-quality temporal contexts. Our offset diversity and quadtree partition are also time-efficient designs in providing diverse contexts.

3. Proposed Method

3.1. Overview

To achieve higher compression ratio, our codec is built on the more flexible conditional coding rather than the residual coding. The framework of our DCVC-DC is illustrated in Fig. 2. It is noted our DCVC-DC is designed for low-delay coding as it can be applied in more scenarios, e.g., real-time communication. As shown in Fig. 2, for coding each frame x_t with frame index t, our coding pipeline contains three core steps: f_{motion}, $f_{Tcontext}$, and f_{frame}. At first, f_{motion} uses optical flow network to estimate the motion vector (MV) v_t, then v_t is encoded and decoded as $\hat{v}_t$. Second, based on $\hat{v}_t$ and the propagated feature F_{t-1} from the previous frame, $f_{Tcontext}$ extracts the motion-aligned temporal context feature C_t. At last, conditioned on C_t, f_{frame} encodes x_t into quantized latent representation $\hat{y}_t$. After entropy coding, the output frame $\hat{x}_t$ is reconstructed via the decoder and frame generator. At the same time, F_t is also generated and propagated to the next frame. It is noted that, our DCVC-DC is based on DCVC-HEM [29]. When compared with DCVC-HEM, this paper redesigns the modules to exploit Diverse Contexts from both temporal (Section 3.2 and 3.3) and spatial (Section 3.4) dimensions.

3.2. Hierarchical Quality Structure

Traditional codec widely adopts hierarchical quality structure, where frames are assigned into different layers and then use different QPs (quantization parameters). This design originates from scalable video coding [48] but also improves the performance for general low delay coding from two aspects. One is periodically improving the quality can alleviate the error propagation, as shown in Fig. 3. During the inter prediction, the high-quality reference frames enable the codec to find the more accurate MV during motion estimation. Meanwhile, the motion compensated prediction is also high-quality, leading to smaller prediction error. Another aspect is that, powered by multiple reference frame selection and weighted prediction mechanisms, the prediction combinations from the nearest reference frame and long-range high-quality reference frame are more diverse. The work [30] investigates many settings on frame quality and reference frame selection, and concludes that the hierarchical quality structure with referencing both the nearest frame and farther high-quality frame achieves the best performance.

Inspired by the success in traditional codec, we are thinking whether we can equip NVC with the hierarchical quality structure and let NVC also enjoy the benefits. Considering the recent neural codecs [11,29] also support variable bitrate in single model, one straightforward solution is following the traditional codec and directly assigning hierarchical QPs during the inference of NVC. However, not like traditional codec uses well-defined rules to perform motion estimation and motion compensation (MEMC), NVC uses neural net-

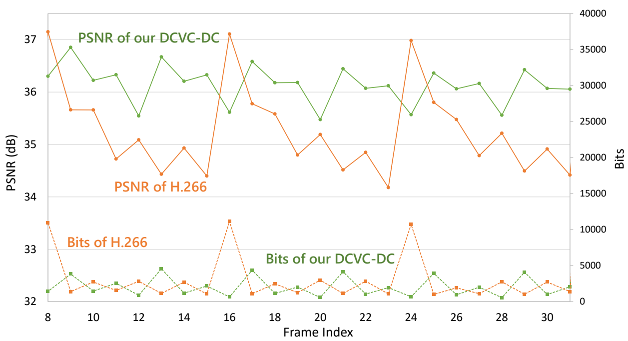

Figure 3. Hierarchical quality structure in H.266-VTM 17.0 and our NVC. This example is from *BasketballPass* video from HEVC D dataset. The average bpp (bits per pixel) and PSNR of H.266 are 0.056 and 35.10. Our DCVC-DC is with 0.045 and 36.13.

works and MEMC is often in feature domain. For NVC, the advantage of such design is that it is automatic learned and has larger potential to achieve better performance. The disadvantage is that it has wore robustness and generalization ability for out-of-distribution quality pattern. Thus, if we directly feed NVC the hierarchical QPs during the testing like [22], it may not well adapt to the hierarchical quality pattern, and the MEMC may get sub-optimal performance.

To this end, we propose guiding the NVC to learn the hierarchical quality pattern during the training. Specifically, we add a weight w_t for each frame in the rate-distortion loss, as shown in Fig. 2. The setting of w_t follows the hierarchical structure. Powered by this hierarchical distortion loss, both high-quality output frame $\hat{x}_t$ and feature F_t containing many high-frequency details are periodically generated. They are very helpful for improving the MEMC effectiveness and then alleviate the error propagation problem that many other NVCs suffer from. In addition, via the cascaded training across multiple frames, the feature propagation chain is formed. The high-quality contexts which are vital for the reconstruction of the following frames are automatically learned and kept in long range. Thus, for the encoding of x_t, F_{t-1} not only contains the short-term contexts extracted from x_{t-1}, but also provides long-term and continually-updated high-quality contexts from many previous frames. Such diverse F_{t-1} helps further exploit the temporal correlation across many frames and then boost the compression ratio. Fig. 3 also shows the quality pattern of our NVC. We can see that our DCVC-DC achieves better average quality while with smaller bit cost than H.266.

3.3. Group-Based Offset Diversity

Due to the various motions between frames, directly using the unprocessed F_{t-1} without motion alignment is hard for codec to capture temporal correspondence. Therefore, we follow existing NVCs and use optical flow network to extract motion aligned temporal context C_t via $f_{Tcontext}(F_{t-1}, \hat{v}_t)$, where $\hat{v}_t$ is the decoded MV. However,

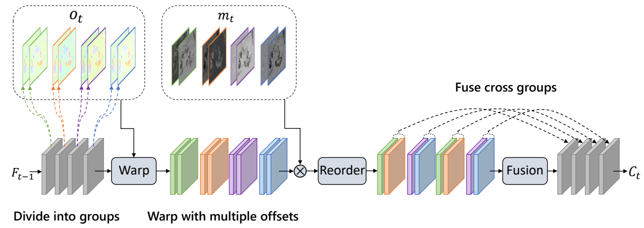

Figure 4. Cross-group fusion module. In this example, the group number G is 4 and the offset number N of each group is 2.

in most existing NVCs, $f_{Tcontext}$ is only the warping operation with the single MV. Such single motion-based alignment is not robust to complex motions or occlusions. The works [9, 54, 58] show deformable alignment gets better results for video restoration as each location has multiple offsets to capture temporal correspondence. So recently it is also applied into NVC [23]. However, the training of deformable alignment is not stable and the overflow of offsets degrades the performance [8, 58]. In addition, the number of offsets is limited to the size of deformable convolutional kernel. Thus, this paper adopts the more flexible design called offset diversity [8]. Meanwhile, the decoded MV is used as the base offset to stabilize the training as [9].

As shown in Fig. 2, our $f_{Tcontext}$ consists of two core sub-modules: offset prediction and cross-group fusion. At first, offset prediction uses the decoded MV $\hat{v}_t$ to predict the residual offsets d_t, where $\hat{x}_{t-1}$ and F_{t-1} are also warped and fed as the auxiliary information. The d_t adds the base offset $\hat{v}_t$ to obtain the final offsets o_t. At the same time, offset prediction also generates the modulation mask m_t which can be regarded as an attention that reflects the confidence of offsets. It is noted that F_{t-1} is divided into G groups along the channel dimension, and each group has separate N offsets. Thus, there are a total of $G \times N$ offsets learned. The diverse offsets are complementary to each other, and help codec cope with complex motion and occlusion.

In addition, motivated by the channel shuffle operation [62] which improves the information flow in the CNN backbone, we customize a group-level interaction mechanism to further tap the potential of offset diversity for NVC. In particular, after warping each group with multiple offsets and applying the corresponding masks, we will reorder all groups before the fusion, as shown in Fig. 4. If using g_i^j to represent the i-th group warped with its j-th offset, the features before reordering are $g_0^0, ..., g_0^{N-1}, g_1^0, ..., g_1^{N-1}, \cdots, g_{G-1}^0, ..., g_{G-1}^{N-1}$, where the offset order is primary and group order is secondary. Then we reorder them as $g_0^0, ..., g_{G-1}^0, g_1^1, ..., g_{G-1}^1, \cdots, g_0^{N-1}, ..., g_{G-1}^{N-1}$, where the group order is primary instead. The following fusion operation will fuse every N contiguous groups into one group. Therefore, during this process, the group reordering enables more cross-group interactions without increasing

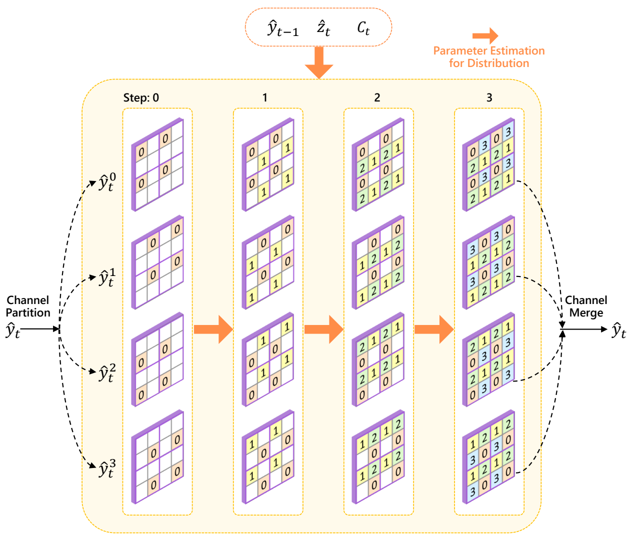

Figure 5. Entropy coding with quadtree partition. The number means the coding order index. During the 4 coding steps, the $\hat{y}_{t-1}$ from previous frame, hyper prior $\hat{z}_t$, and temporal context C_t are also used for entropy modelling.

complexity. This design also enjoys the similar benefit with the weighted prediction from different reference frames in traditional codec. Via the cross-group fusion, more diverse combinations in extracting temporal contexts from different groups are introduced and further improve the effectiveness of offset diversity.

3.4. Quadtree Partition-Based Entropy Coding

After obtaining temporal context C_t via our offset diversity module, the input frame x_t will be encoded and quantized as $\hat{y}_t$, conditioned on C_t, as shown in Fig. 2. We need to estimate the probability mass function (PMF) of $\hat{y}_t$ for its arithmetic coding. In this process, how to build an accurate entropy model to estimate the PMF of $\hat{y}_t$ is vital for the compression efficiency.

Many neural codecs adopt the auto-regressive model [40] as entropy model. However, it seriously slows down the coding speed. By contrast, the checkerboard model [19] proposes coding the even positions of $\hat{y}_t$ first, and then use them to predict the PMF of the odd positions in parallel. Recently the dual spatial model [29] improves it by utilizing the correlation along channel dimension. However, the neighbours used for entropy modelling in [19, 29] are still limited when compared with auto-regressive model. Thus, inspired from [41, 46], this paper proposes a finer-grained coding manner via the quadtree partition, where diverse spatial contexts are exploited to improve entropy modelling.

As shown in Fig. 5, we first divide $\hat{y}_t$ into four groups along the channel dimension. Then each group is partitioned into non-overlapped 2×2 patches in spatial dimension. The whole entropy coding is divided into four steps, and each step codes the different positions associated with the corresponding indexes in Fig. 5. At 0th step, all positions with index 0 of all patches are coded at the same time. It is noted, for the four groups, the positions with index 0 are different from each other. Thus, for every spatial position, there are one fourth channels (i.e., one group) encoded. In the subsequent 1st, 2nd, and 3rd steps, all positions coded in previous steps are used for predicting the PMF of the positions coded in the current step, and different spatial positions are coded for different groups in each step.

During this process, more diverse neighbours are utilized. If considering the 8 spatial neighbours for a position, the auto-regressive model [40] uses 4 (left, top-left, top, top-right) neighbours for every position if not considering the boundary region. The checkerboard and dual spatial models [19, 29] uses 0 and 4 (left, top, right, bottom) neighbours for the 0th and 1st steps, respectively. By contrast, as shown in Fig. 5, our DCVC-DC uses 0, 4, 4, and 8 neighbours for the four steps, respectively. On average, the neighbour number in DCVC-DC is 2 times of that of [19, 29] and is same with that of auto-regressive model. However, our model is much more time-efficient than auto-regressive model as all positions in each step can be coded in parallel. In addition, our model also exploits the cross-channel correlation, which is like [29] but in a refined way. For example, at the 3rd step, for one specific position of a group, the other channels at the same position are already coded from different groups in previous steps, and they can be used as the contexts for the entropy modelling in this step. This helps further squeeze the redundancy. Overall, our quadtree partition-based solution makes the entropy coding benefit from the finer-grained and diverse contexts, which fully mines the correlation from both spatial and channel dimensions.

3.5. Implementation

Our DCVC-DC is based on DCVC-HEM [29] but focuses on exploiting Diverse Contexts to further boost performance. In addition, we also make the following improvements to obtain better tradeoff between performance and complexity. The first is that, considering depthwise separable convolution [10] can reduce the computation cost while alleviating over-fitting, we widely use it to replace the normal convolution in the basic block design. The second is that we use the unequal channel number settings for features with different resolutions, where the higher resolution feature is assigned with smaller channel number for acceleration. The third is that we move partial quantization operations to higher resolution in the encoder, which helps achieve more precise bit allocation. The harmonization of encoding and quantization also brings some compression ratio improvements. The section 4.3 verifies the effectiveness of these structure optimizations, and the detailed network structures can be found in supplementary materials.

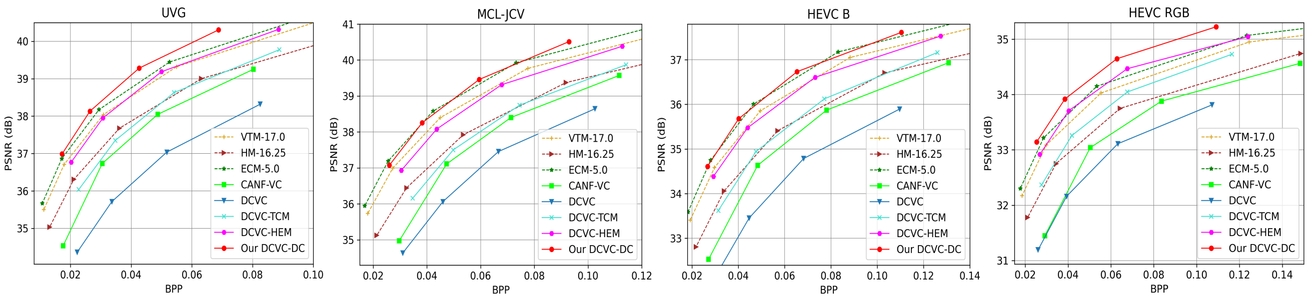

Figure 6. Rate and distortion curves. The comparison is in RGB colorspace measured with PSNR. More results including the corresponding MS-SSIM curves and the comparison in YUV420 colorsapce are in supplementary materials.

Table 1. BD-Rate (%) comparison in RGB colorspace measured with PSNR. The anchor is VTM-17.0.

	UVG	MCL-JCV	HEVC B	HEVC C	HEVC D	HEVC E	HEVC RGB	Average
VTM-17.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HM-16.25	36.4	41.5	38.8	36.0	33.7	44.0	39.4	38.5
ECM-5.0	–10.0	–12.2	–11.5	–13.4	–13.5	–10.9	–11.1	-11.8
CANF-VC [21]	73.0	70.8	64.4	76.2	63.1	118.0	79.9	77.9
DCVC [28]	166.1	121.6	123.2	143.2	98.0	266.1	113.4	147.4
DCVC-TCM [50]	44.1	51.0	40.2	66.3	37.0	82.7	24.4	49.4
DCVC-HEM [29]	1.1	8.6	5.1	22.2	2.4	20.5	-9.9	7.1
Our DCVC-DC	–19.1	–11.3	–12.0	–10.3	–26.1	–18.0	–27.6	–17.8

In particular, our DCVC-DC also already outperforms ECM in YUV420 colorspace. Not like traditional codec needs many hand-crafted changes in designing coding tools for different colorspaces, DCVC-DC only just needs simple adaptions based on a existing model trained for RGB. Without changing the network structure, we only up-sample the UV to use the unified input interface with RGB. Correspondingly, after obtaining the reconstructed frame, it is down-sampled on UV. In addition, our model for YUV420 just needs a simple finetune training based on the model trained for RGB.

4. Experimental Results

4.1. Experimental Settings

Datasets. For training, we follow most existing NVCs and use Vimeo-90k [60]. For testing YUV420 videos, HEVC B∼E [6], UVG [39], and MCL-JCV [57] are used. Their raw format is YUV420, so there is no any change before feeding them to NVC. For testing RGB videos, as these testsets have no RGB format, most existing NVCs use BT.601 (default in FFmpeg) to convert them from YUV420 to RGB. Actually, JPEG AI [2,3] adopts BT.709 because using BT.709 obtains higher compression ratio under similar visual quality. Thus, this paper follows JPEG AI and uses BT.709 for all codecs during testing RGB. It is noted that the relative bitrate comparisons between different codecs are similar in BT.601 and BT.709. The supplementary materials show the results using BT.601. In addition, we follow [29, 50] and also test HEVC RGB dataset [15] when testing RGB videos, and there is no format change as HEVC RGB dataset itself is in RGB format.

Test Conditions. We follow [29, 50] and test 96 frames for each video, and the intra period is set as 32. The low delay encoding setting is used, as the same with most existing works [1, 28, 34]. BD-Rate [5] is used to measure the compression ratio, where negative numbers indicate bitrate saving and positive numbers indicate bitrate increase.

Our benchmarks include HM [20] and VTM [55] which represent the best H.265 and H.266 encoder, respectively. In particular, we also compare with ECM [14] which is the prototype of next generation traditional codec. For the codec setting, we follow [29, 50] and further use 10-bit as the intermediate representation when testing RGB, which leads to better compression ratio for the three traditional codecs. The detailed settings are shown in supplementary materials. As for the NVC benchmarks, we compare with the recent SOTA models including CANF-VC [21], DCVC [28], DCVC-TCM [50], and DCVC-HEM [29].

Model Training. We adopt the multi-stage training strategy as [29, 50]. Our model also supports variable bitrate in single model [29], so different λ values are used in different optimization steps. We follow [29] and use 4 λ values (85, 170, 380, 840). But different from [29] using constant distortion weight in the loss, this paper propose using hierarchical weight setting on w_t for the distortion term (the

Table 2. BD-Rate (%) comparison in RGB colorspace measured with MS-SSIM. The anchor is VTM-17.0.

	UVG	MCL-JCV	HEVC B	HEVC C	HEVC D	HEVC E	HEVC RGB	Average
VTM-17.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HM-16.25	31.1	38.8	36.6	35.2	33.0	41.1	36.6	36.1
ECM-5.0	–9.1	–11.1	–10.2	–11.7	–11.0	–9.9	–9.8	–10.4
CANF-VC [21]	46.5	26.0	43.5	30.9	17.9	173.0	57.7	56.5
DCVC [28]	64.9	27.5	54.4	39.7	15.2	210.4	51.3	66.2
DCVC-TCM [50]	1.0	–10.8	–11.7	–15.2	–29.0	16.7	–22.2	–10.2
DCVC-HEM [29]	–25.2	–36.3	–38.0	–38.3	–48.1	–25.8	–43.6	–36.5
Our DCVC-DC	–32.6	–44.8	–47.8	–49.8	–58.2	–45.8	–54.4	–47.6

Table 3. BD-Rate (%) comparison in YUV420 colorspace measured with PSNR. The anchor is VTM-17.0.

	UVG	MCL-JCV	HEVC B	HEVC C	HEVC D	HEVC E	Average
VTM-17.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HM-16.25	36.7	42.5	39.2	33.3	30.0	40.7	37.1
ECM-5.0	–10.6	–13.7	–12.6	–14.7	–14.9	–12.1	–13.1
Our DCVC-DC	–17.8	–12.0	–10.8	–12.4	–28.5	–20.4	–17.0

whole loss is defined in Fig. 2). Considering our training set Vimeo-90k only has 7 frames for each video, we refer traditional codec setting and set the pattern size as 4. The w_t settings for 4 consecutive frames are (0.5, 1.2, 0.5, 0.9).

4.2. Comparisons with Previous SOTA Methods

RGB colorspace. Table 1 and 2 show the BR-rate comparison using RGB videos in terms of PSNR and MS-SSIM, respectively. From Table 1, we find our codec achieves significant compression ratio improvement over VTM on every dataset, and there is an average of 17.8% bitrate saving. By contrast, the other neural codecs are still worse than VTM. If using DCVC-HEM as anchor, our average bitrate saving is 23.5%. In addition, our DCVC-DC also outperform ECM from Table 1. If using ECM as anchor, an average of 6.4% bitrate saving is achieved.

Fig. 6 shows the rate-distortion curves. From the curves, we can see our DCVC-DC achieves the SOTA compression ratio in wide bitrate range. When using MS-SSIM as quality metric, our DCVC-DC shows larger improvement. As shown in Table 2, DCVC-DC has an average of 47.6% bitrate saving over VTM. By contrast, the corresponding number of ECM over VTM is only 10.4%.

It is noted that Table 1 and 2 use the RGB video with BT.709 conversion. If with BT.601 conversion, the relative bitrate saving is similar with that in BT.709. For example, with BT.601 conversion, DCVC-DC over VTM has an average of 18.0% bitrate saving in terms of PSNR. More results with BT.601 can be seen in supplementary materials.

YUV420 colorspace. Actually traditional codecs are mainly optimized in YUV420. Thus, the comparison in YUV420 is also very important for evaluating the progress of NVC over traditional codec. The corresponding results are shown in Table 3. The numbers in this table are calculated using the weighted PSNR for the three components of YUV. The weights are (6,1,1)/8, which are consisted with that in standard committee [53]. As most NVCs have no corresponding released models for YUV420, Table 3 only reports the numbers of our NVC. We can see that DCVC-DC has an average of 17.0% bitrate saving over VTM. If only considering the Y component, an average 15.3% bitrate saving is achieved over VTM. Better yet, our DCVC-DC also outperforms ECM in YUV420 on average, as shown in Table 3. This is an important milestone in the development of NVC. It is noted our codec uses the same network structure for both RGB and YUV420 colorspaces, where only different finetunings are used during training. This shows the simplicity and strong extensibility of our codec on the optimization for different input colorspaces.

4.3. Ablation Study

To verify the effectiveness of each component, we conduct comprehensive ablation studies. For simplification, the HEVC datasets in RGB colorspace are used here. The average BD-Rate in terms of PSNR is shown.

Diverse Contexts. Table 4 shows the study on the effectiveness of diverse contexts. First, from the comparison between M_e and M_d, we can see that the BD-rate is reduced from 21.3% to 14.7%. This large difference shows the substantial coding gain of our qaudtree partition-based entropy coding, and verifies the advantages of diverse spatial and channel contexts via finer-grained partition.

Table 4. Ablation Study on Diverse Contexts.

	M_a	M_b	M_c	M_d	M_e
Hierarchical quality structure	✓				
Offset diversity w/ cross-group	✓	✓			
Offset diversity w/o cross-group [8]			✓		
Quadtree partition based model	✓	✓	✓	✓	
Dual spatial model [29]					✓
BD-Rate(%)	0.0	8.4	12.1	14.7	21.3

From temporal dimension, we also design hierarchical quality structure and offset diversity with cross-group interaction. In Table 4, based on M_d, we first test the original offset diversity [8] without cross-group interaction, i.e., removing the reorder operation in Fig. 4. However, it (M_c) only brings 2.6% BD-rate difference. By contrast, powered by our cross-group interaction, the potential of offset diversion is fully tapped, and M_b reduces the BD-rate number by 6.3% over M_d. At last, based on M_b, we evaluate the hierarchical quality structure, i.e., M_a. The 8.4% gap shows learning high-quality contexts brings large benefits to the mining of temporal correlation across many frames.

Structure optimization. Although our codec learns utilizing diverse contexts in efficient manner, we still purse better tradeoff between compression ratio and computational cost. Therefore, we further optimize our model in network structure. Table 5 shows the study. Based on the M_a (same with that in Table 4), we first implement the depthwise separable convolution into codec. M_h shows widely using depthwise separable convolution to replace the normal convolution not only significantly reduces the MACs, but also brings some compression ratio improvements.

The second acceleration is that we use the unequal channel number settings. Not like many existing NVCs use the same channel number for features with different resolutions, we propose assigning the larger number for low resolution feature to increase the latent representation capacity while using the smaller number for the high resolution feature to accelerate model. The performance of M_g verifies the effectiveness of our improvement. In addition, many existing NVCs perform the quantization at the low-resolution latent representation after encoding. To achieve more precise rate adjustment, this paper moves partial quantization operations to the higher resolution in the encoder. M_f shows the integration of encoding and quantization brings some BD-rate improvements with negligible MAC change.

4.4. Complexity

The complexity comparison is shown in Table 6. We find the MACs of our DCVC-DC are reduced by 19.4% when compared with DCVC-HEM [29]. However, the actual encoding and decoding time is higher. This is because currently the computational density of depthwise convolution is not as high as normal convolution under the same MAC condition. But through the customized optimization [35], it can be further accelerated in the future. From another perspective, considering that our DCVC-DC has 23.5% bitrate saving over previous SOTA DCVC-HEM [29], such increase degree in running time is a price worth paying. By contrast, ECM brings 13.1% (Table 3) improvement over its predecessor VTM, but the encoding complexity is more than 4 times [49] of VTM.

Table 5. Ablation Study on Structure Optimization.

	M_f	M_g	M_h	M_a
Quant at high resolution	✓			
Unequal channel setting	✓	✓		
Depthwise separable conv	✓	✓	✓	
MACs	2642G	2642G	2939G	3456G
BD-Rate (%)	0.0	1.1	2.4	3.5

Table 6. Complexity comparison.

	MACs	Encoding Time	Decoding Time
DCVC-HEM [29]	3279G	890ms	652ms
Our DCVC-DC	2642G	1005ms	765ms

Note: Tested on NVIDIA 2080TI with using 1080p as input.

5. Conclusion and Limitation

In this paper, we have presented how to utilize diverse contexts to further boost NVC. From temporal dimension, the model is guided to extract long-term and yet high-quality contexts to alleviate error propagation and exploit long range correlation. The offset diversity with cross-group interaction provides complementary motion alignments to handle complex motion. From spatial dimension, the fine-grained quadtree-based partition is proposed to increase spatial context diversity. Powered by our techniques, the compression ratio of NVC has been pushed to new height. Our DCVC-DC has surpassed ECM in both RGB and YUV420 colorspaces, which is an important milestone in the development of NVC.

During the training, to learn the hierarchical quality pattern, we still use the fixed distortion weights which are similar with those in traditional codec. This may not be the best choice for NVC. Actually, reinforcement learning is good at solving such kind of time series weight decision problem. In the future, we will investigate utilizing reinforcement learning to help NVC make better weight decision with considering the temporal dependency.

References

[1] Eirikur Agustsson, David Minnen, Nick Johnston, Johannes Balle, Sung Jin Hwang, and George Toderici. Scale-space flow for end-to-end optimized video compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8503–8512, 2020. 1, 2, 6

[2] E. Alshina, J. Ascenso, T. Ebrahimi, F. Pereira, and T. Richter. [AHG 11] Brief information about JPEG AI CfP status. In *JVET-AA0047*, 2022. 6

[3] Anchors · JPEG-AI MMSP Challenge. `https://jpegai.github.io/7-anchors/`, 2022. Accessed: 2022-11-02. 6

[4] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. *6th International Conference on Learning Representations, ICLR*, 2018. 2

[5] Gisle Bjontegaard. Calculation of average PSNR differences between RD-curves. *VCEG-M33*, 2001. 6

[6] Frank Bossen et al. Common test conditions and software reference configurations. In *JCTVC-L1100*, 2013. 6

[7] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J Sullivan, and Jens-Rainer Ohm. Overview of the versatile video coding (VVC) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):3736–3764, 2021. 1, 2

[8] Kelvin CK Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Understanding deformable alignment in video super-resolution. In *Proceedings of the AAAI conference on artificial intelligence*, 2021. 2, 4, 8

[9] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. BasicVSR++: Improving video super-resolution with enhanced propagation and alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5972–5981, 2022. 4

[10] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 2, 5

[11] Ze Cui, Jing Wang, Shangyin Gao, Tiansheng Guo, Yihui Feng, and Bo Bai. Asymmetric gained deep image compression with continuous rate adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10532–10541, 2021. 3

[12] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural safety*, 31(2):105–112, 2009. 1, 2

[13] Abdelaziz Djelouah, Joaquim Campos, Simone Schaub-Meyer, and Christopher Schroers. Neural inter-frame compression for video coding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 1

[14] ECM-5.0. `https://vcgit.hhi.fraunhofer.de/ecm/ECM`, 2022. Accessed: 2022-11-02. 6

[15] D Flynn, K Sharman, and C Rosewarne. Common test conditions and software reference configurations for hevc range extensions, document jctvc-n1006. *Joint Collaborative Team Video Coding ITU-T SG*, 16, 2013. 6

[16] Yarin Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016. 1, 2

[17] Bernd Girod, Eckehard G Steinbach, and Niko Faerber. Comparison of the H. 263 and H. 261 video compression standards. In *Standards and Common Interfaces for Video Information Systems: A Critical Review*, 1995. 1

[18] Zongyu Guo, Zhizheng Zhang, Runsen Feng, and Zhibo Chen. Causal contextual prediction for learned image compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(4):2329–2341, 2021. 2

[19] Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. Checkerboard context model for efficient learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14771–14780, 2021. 2, 5

[20] HM-16.25. `https://vcgit.hhi.fraunhofer.de/jvet/HM/`, 2022. Accessed: 2022-11-02. 6

[21] Yung-Han Ho, Chih-Peng Chang, Peng-Yu Chen, Alessandro Gnutti, and Wen-Hsiao Peng. Canf-vc: Conditional augmented normalizing flows for video compression. *European Conference on Computer Vision*, 2022. 1, 6, 7

[22] Zhihao Hu, Guo Lu, Jinyang Guo, Shan Liu, Wei Jiang, and Dong Xu. Coarse-to-fine deep video coding with hyperprior-guided mode prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5921–5930, 2022. 4

[23] Zhihao Hu, Guo Lu, and Dong Xu. FVC: A new framework towards deep video compression in feature space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1502–1511, 2021. 4

[24] Yan Huang, Li Song, and Ebroul Izquierdo. CNN accelerated intra video coding, where is the upper bound? In *2019 Picture Coding Symposium (PCS)*, pages 1–5. IEEE, 2019. 1

[25] Shuai Huo, Dong Liu, Li Li, Siwei Ma, Feng Wu, and Wen Gao. Towards hybrid-optimization video coding. *arXiv preprint arXiv:2207.05565*, 2022. 1, 2, 3

[26] Jun-Hyuk Kim, Byeongho Heo, and Jong-Seok Lee. Joint global and local hierarchical priors for learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5992–6001, 2022. 2

[27] Théo Ladune, Pierrick Philippe, Wassim Hamidouche, Lu Zhang, and Olivier Déforges. Conditional coding for flexible learned video compression. In *Neural Compression: From Information Theory to Applications – Workshop @ ICLR*, 2021. 1, 2

[28] Jiahao Li, Bin Li, and Yan Lu. Deep contextual video compression. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 2, 6, 7

[29] Jiahao Li, Bin Li, and Yan Lu. Hybrid spatial-temporal entropy modelling for neural video compression. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1503–1511, 2022. 1, 2, 3, 5, 6, 7, 8

[30] Chong Soon Lim, SMT Naing, V Wahadaniah, and X Jing. Reference lists for B pictures under low delay constraints.

document JCTVC-D093, ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC), Daegu, Korea, 2011. 3

[31] Jianping Lin, Dong Liu, Houqiang Li, and Feng Wu. M-LVC: multiple frames prediction for learned video compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2

[32] Haojie Liu, Ming Lu, Zhan Ma, Fan Wang, Zhihuang Xie, Xun Cao, and Yao Wang. Neural video coding using multiscale motion compensation and spatiotemporal context model. *IEEE Transactions on Circuits and Systems for Video Technology*, 2020. 1

[33] Jerry Liu, Shenlong Wang, Wei-Chiu Ma, Meet Shah, Rui Hu, Pranaab Dhawan, and Raquel Urtasun. Conditional entropy coding for efficient video compression. In *European Conference on Computer Vision*, pages 453–468. Springer, 2020. 1, 2

[34] Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao. DVC: an end-to-end deep video compression framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11006–11015, 2019. 1, 2, 6

[35] Gangzhao Lu, Weizhe Zhang, and Zheng Wang. Optimizing depthwise separable convolution operations on gpus. *IEEE Transactions on Parallel and Distributed Systems*, 33(1):70–87, 2021. 8

[36] Guo Lu, Xiaoyun Zhang, Wanli Ouyang, Li Chen, Zhiyong Gao, and Dong Xu. An end-to-end learning framework for video compression. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 1

[37] Wufei Ma, Jiahao Li, Bin Li, and Yan Lu. Uncertainty-aware deep video compression with ensembles, 2021. 1, 2

[38] Fabian Mentzer, George Toderici, David Minnen, Sung-Jin Hwang, Sergi Caelles, Mario Lucic, and Eirikur Agustsson. Vct: A video compression transformer. *arXiv preprint arXiv:2206.07307*, 2022. 1, 2

[39] Alexandre Mercat, Marko Viitanen, and Jarno Vanne. UVG dataset: 50/120fps 4k sequences for video codec analysis and development. In *Proceedings of the 11th ACM Multimedia Systems Conference*, pages 297–302, 2020. 6

[40] David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. *Advances in neural information processing systems*, 31, 2018. 2, 5

[41] Ken M Nakanishi, Shin-ichi Maeda, Takeru Miyato, and Daisuke Okanohara. Neural multi-scale image compression. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part VI 14*, pages 718–732. Springer, 2019. 5

[42] Jonathan Pfaff, Alexey Filippov, Shan Liu, Xin Zhao, Jianle Chen, Santiago De-Luxan-Hernandez, Thomas Wiegand, Vasily Rufitskiy, Adarsh Krishnan Ramasubramonian, and Geert Van der Auwera. Intra prediction and mode coding in VVC. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):3834–3847, 2021. 1

[43] Reza Pourreza, Hoang Le, Amir Said, Guillaume Sautiere, and Auke Wiggers. Boosting neural video codecs by exploiting hierarchical redundancy. *arXiv preprint arXiv:2208.04303*, 2022. 2

[44] Yichen Qian, Ming Lin, Xiuyu Sun, Zhiyu Tan, and Rong Jin. Entroformer: A transformer-based entropy model for learned image compression. *arXiv preprint arXiv:2202.05492*, 2022. 2

[45] Yichen Qian, Zhiyu Tan, Xiuyu Sun, Ming Lin, Dongyang Li, Zhenhong Sun, Hao Li, and Rong Jin. Learning accurate entropy model with global reference for image compression. *arXiv preprint arXiv:2010.08321*, 2020. 2

[46] Scott Reed, Aäron Oord, Nal Kalchbrenner, Sergio Gómez Colmenarejo, Ziyu Wang, Yutian Chen, Dan Belov, and Nando Freitas. Parallel multiscale autoregressive density estimation. In *International Conference on Machine Learning*, pages 2912–2921. PMLR, 2017. 5

[47] Oren Rippel, Alexander G Anderson, Kedar Tatwawadi, Sanjay Nair, Craig Lytle, and Lubomir Bourdev. ELF-VC: Efficient learned flexible-rate video coding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14479–14488, October 2021. 1, 2

[48] Heiko Schwarz, Detlev Marpe, and Thomas Wiegand. Analysis of hierarchical B pictures and MCTF. In *2006 IEEE International Conference on Multimedia and Expo*, pages 1929–1932. IEEE, 2006. 3

[49] Vadim Seregin, Jie Chen, Fabrice Leannec, and Kai Zhang. JVET AHG report: ECM software development (AHG6). In *JVET-AA0006*, 2022. 1, 8

[50] Xihua Sheng, Jiahao Li, Bin Li, Li Li, Dong Liu, and Yan Lu. Temporal Context Mining for Learned Video Compression. *IEEE Transactions on Multimedia*, 2022. 1, 2, 6, 7

[51] Hui Su, Mingliang Chen, Alexander Bokov, Debargha Mukherjee, Yunqing Wang, and Yue Chen. Machine learning accelerated transform search for av1. In *2019 Picture Coding Symposium (PCS)*, pages 1–5. IEEE, 2019. 1

[52] Hui Su, Chi-Yo Tsai, Yunqing Wang, and Yaowu Xu. Machine learning accelerated partition search for video encoding. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2661–2665. IEEE, 2019. 1

[53] Gary J Sullivan and Jens-Rainer Ohm. Meeting report of the fourth meeting of the joint collaborative team on video coding (jct-vc), daegu, kr, 20–28 january 2011. *Document JCTVC-D500, Daegu, KR*, 2011. 7

[54] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. TDAN: Temporally-deformable alignment network for video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3360–3369, 2020. 4

[55] VTM-17.0. `https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/`, 2022. Accessed: 2022-11-02. 6

[56] Guo-Hua Wang, Jiahao Li, Bin Li, and Yan Lu. EVC: Towards Real-Time Neural Image Compression with Mask Decay. In *International Conference on Learning Representations*, 2023. 2

[57] Haiqiang Wang, Weihao Gan, Sudeng Hu, Joe Yuchieh Lin, Lina Jin, Longguang Song, Ping Wang, Ioannis Katsavounidis, Anne Aaron, and C-C Jay Kuo. MCL-JCV: a JND-based

H. 264/AVC video quality assessment dataset. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 1509–1513. IEEE, 2016. 6

[58] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. EDVR: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 4

[59] Chao-Yuan Wu, Nayan Singhal, and Philipp Krahenbuhl. Video compression through image interpolation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 416–431, 2018. 1

[60] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision (IJCV)*, 127(8):1106–1125, 2019. 6

[61] Ren Yang, Fabian Mentzer, Luc Van Gool, and Radu Timofte. Learning for video compression with recurrent autoencoder and recurrent probability model. *IEEE Journal of Selected Topics in Signal Processing*, 15(2):388–401, 2021. 1, 2

[62] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018. 4