Augmented Deep Contexts for Spatially Embedded Video Coding

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

This supplementary material document provides additional details of our proposed Spatially Embedded Video Codec (SEVC). The remainder of the supplementary material is divided into three parts. Section A gives the configuration of the traditional codec—VTM-13.2 [1]. Section B gives the detailed network architectures of our proposed modules. Section C gives the derivation of Euqation (1). Section D provides additional comparison results.

A. Configuration of the Traditional Codec

When testing traditional codec—VTM-13.2 [1], the input video sequences are in YUV444 format to achieve a better compression ratio [3, 4, 8]. The YUV444 video sequences are converted from the RGB video sequences, which are used as the inputs of NVCs. The configuration parameters for encoding each video are as:

- EncoderAppStatic

  -c encoder_lowdelay_vtm.cfg

  –InputFile={*Input File Path*}

  –InputBitDepth=8

  –OutputBitDepth=8

  –OutputBitDepthC=8

  –InputChromaFormat=444

  –FrameRate={*Frame Rate*}

  –DecodingRefreshType=2

  –FramesToBeEncoded=96

  –IntraPeriod={*Intra Period*}

  –SourceWidth={Width}

  –SourceHeight={Height}

  –QP={QP}

  –Level=6.2

  –BitstreamFile={Bitsteam File Path}

  –ReconFile={Output File Path}

- DecoderAppStatic

  -b {Bitsteam File Path}

  -o {Reconstruction File Path}

B. Network Architechture

Our SEVC is implemented based on DCVC-DC [4] but focuses on exploiting additional spatial references for augmenting the contexts and latent prior.

Motion and Feature Co-Augmentation. As shown in Figure 4, the Motion and Feature Co-Augmentation (MFCA) module progressively improves the quality of the base MVs and the spatial feature through several Augment Stages. Figure a shows the architecture of one Augment Stage. It takes two steps to augment the base MVs $\bar{v}_i^l$ and spatial feature $\bar{F}_i^l$ within one Augment Stage: Firstly, base MVs $\bar{v}_i^l$, spatial feature $\bar{F}_i^l$, and temporal feature $\hat{F}_{t-1}^l$ are fed into an Augment Unit to generate augmented base MVs $\bar{v}_{i+1}^l$. Secondly, the augmented base MVs $\bar{v}_{i+1}^l$ leads a better alignment of $\hat{F}_{t-1}^l$ and the aligned $\hat{F}_{t-1}^l$ are fed into another Augment Unit with spatial feature $\bar{F}_i^l$ to generate augmented spatial feature $\bar{F}_{i+1}^l$. Figure b shows the architecture of one Augment Unit. This example is the Augment Unit for motion augmentation in the largest scale, where $N^l = 48$. Two convolution layers with a stride equal to 2 are used to reduce the resolution and two subpixel layers [10] are used to upsample the residual back to the original resolution.

Spatial-Guided Latent Prior Augmentation The proposed latent prior $\bar{y}_t$ is generated by adding the residual queried from multiple temporal latent representations $\hat{y}_{t-1}, \hat{y}_{t-2}, \hat{y}_{t-3}$ to the upsampled spatial latent $\hat{y}_t^b$. To implement this, two subpixel layers are first used to upsample the spatial latent $\hat{y}_t^b$. The upsampled $\hat{y}_t^b$ and temporal latent representations $\hat{y}_{t-1}, \hat{y}_{t-2}, \hat{y}_{t-3}$ are concatenated and fed into several Residual Swin Transformer Blocks (RSTBs) [7, 9] to generate the residual. Within each RSTB, there are several Swin Transformer Layers (STLs) that utilize 3D window partitions to capture correlation across the

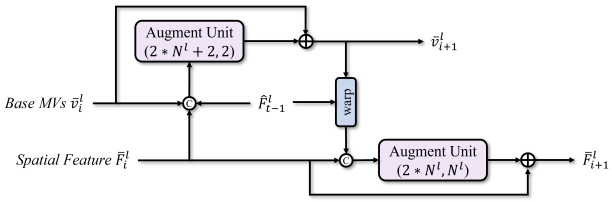

Figure a. The network architechture of the ith Augment Stage. The numbers in an Augment Unit refer to the number of input channels and number of output channels. N^l refers to the number of channels in the lth scale.

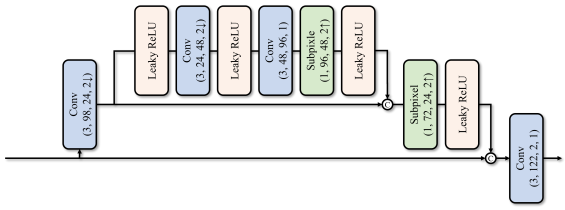

Figure b. The network architecture of the Augment Unit. The numbers in a Conv block refer to the kernel size, number of input channels, number of output channels, and stride. This example is the Augment Unit for motion augmentation in the largest scale.

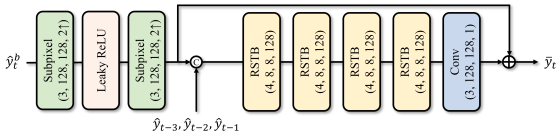

Figure c. The network architecture of the latent prior generation. The numbers in a Residual Swin Transformer Block (RSTB) [7] refer to the depth, head number, window size, and the embedding dimension.

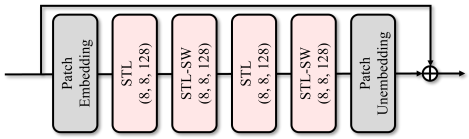

Figure d. The network architecture of the RSTB. The numbers in a Swin Transformer Layer (STL) refer to the head number, window size, and the embedding dimension. STL-SW indicates STL with shifted window partitions.

spatial and temporal dimensions. We set the head number to 8, the window size to 8, and the embedding dimension to 128. A Swin Transformer Layer with shifted window partitions is denoted as STL-SW.

In our SEVC, the Transformers are calculated on low-resolution latent representations, which will not bring too much computation cost. Table a gives the complexity comparison when different numbers of temporal latent representations are used for augmentation. It can be observed that introducing more temporal latent representations only results in a linear complexity increase.

Table a. Complexity Comparison.

ΔT	0	1	2	3	4
MACs	50G	75G	100G	125G	150G

[1] Tested on 1080p sequences.

C. Derivation of Equation (1)

Considering downsampling the input full-resolution video x to the low-resolution (LR) base video x^b, with their respective sources denoted as X and X^b. From a perspective of information theory [2], the mutual information between the source of x^b and x is

$$I(X^b; X) = H(X^b) - H(X^b|X), \tag{a}$$

where X^b and X denote the source of the base video and original video. Given that x^b is fully derived from x through a fixed downsampling algorithm, The conditional probability $p(x^b|x)$ is constant to 1. Thereby, the conditional entropy

$$H(X^b|X) = -\sum_x p(x) \sum_{x^b} p(x^b|x) \log p(x^b|x) \tag{b}$$

is constant to 0. Therefore, we can follow

$$I(X^b; X) = H(X^b). \tag{c}$$

Furthermore, the mutual information can be expressed equivalently as

$$I(X^b; X) = I(X; X^b) = H(X) - H(X|X^b). \tag{d}$$

Equation (c) and (d) follow directly from there with

$$H(X) = H(X^b) + H(X|X^b). \tag{e}$$

Equation (e) is not proposed by us, but is a conclusion well known in scalable coding and a goal that everyone wants to approach. However, it is non-trivial to verify it for complex signals such as videos. Nevertheless, the superior performance of our SEVC makes further explorations for this conclusion.

D. Additional Results

D.1. Results on RGB PSNR with BT.709

When testing RGB videos, we use FFmpeg to convert YUV420 videos to RGB videos, where BT.601 is employed to implement the conversion. However, BT.709 is used in [4, 5] for a higher compression ratio under a similar visual quality. Thus we provide additional results with BT.709 on four 1080p datasets. We focus on high-resolution videos because a 4x downsampling process is conducted in our spatially embedded codec, and the spatial references with too small resolution are meaningless.

Figure e and Table b give the RD curves and BD-Rate

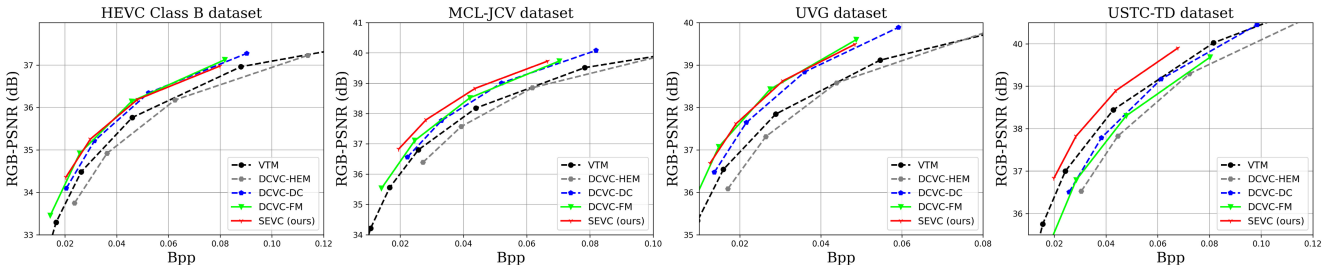

Figure e. Rate and distortion curves on four 1080p datasets. The Intra Period is –1 with 96 frames.

Table b. BD-Rate (%) comparison for RGB PSNR with BT.709. The Intra Period is –1 with 96 frames. The anchor is VTM-13.2 LDB.

	HEVC B	MCL-JCV	UVG	USTC-TD	Average
DCVC-HEM [3]	13.4	11.4	12.5	27.1	16.1
DCVC-DC [4]	–13.4	–13.5	–20.6	11.8	–8.9
DCVC-FM [5]	**–18.1**	–15.9	**–27.9**	23.1	–9.7
SEVC (ours)	–16.5	**–24.5**	–27.0	**–14.5**	**–20.6**

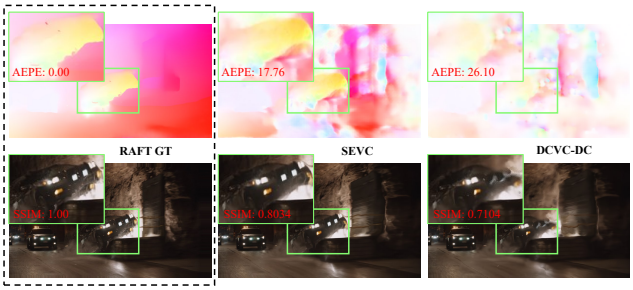

Figure f. Comparison of reconstructed MVs and warp prediction of DCVC-DC and ours. Fewer AEPE scores indicate higher quality MVs and higher SSIM scores demonstrate better alignment.

comparisons for four 1080p datasets with BT.709. Compared to PSNR with BT.601 shown in Figure 10, although PSNR with BT.709 is significantly higher than PSNR with BT.601 under the same bpp, the relative bitrate savings compared to VTM are similar. When using BT.601 and compared to VTM, DCVC-DC achieves an average bitrate saving of 8.3%, DCVC-FM achieves 6.2%, and SEVC achieves 23%. When using BT.709 and compared to VTM, DCVC-DC averages 8.9% savings, DCVC-FM averages 9.7%, and SEVC averages 20.6%. The slight performance degradation is attributed to the fact that the selected train set is not constructed by BT.709 conversion.

D.2. Visualization of MVs and contexts

In order to intuitively demonstrate the improvement in MV quality brought by our progressively augmentation, We compare the MVs of DCVC-DC and ours using the pseudo ground truth generated by RAFT [11]. Average Endpoint Error (AEPE) is used to evaluate MVs quality and Structural Similarity (SSIM) is used to measure warp quality of the MVs. As shown in Figure f, both in subjective perception and objective metrics, our MV is better than that of DCVC-DC in large motion areas whose MVs are greater than 15 pixels.

As shown in Table 3, our SEVC performs much better than DCVC-DC in sequences with large motions and significant emerging objects. There are two main reasons for this: On the one hand, the base MVs progressively augmented by our proposed MFCA module have a higher quality than the reconstructed MVs in DCVC-DC, which improves the utilization of temporal references. On the other hand, the augmented spatial feature can provide an additional description for regions with emerging objects that are not well described by temporal references.

As shown in Figure g, Figure h, and Figure i, the augmented MVs in our SEVC have a higher warp PSNR and a better subjective quality compared to reconstructed MVs in DCVC-DC. However, the residuals are still large in regions where new objects appear (marked in red boxes), indicating that the temporal references are not rich enough to describe the emerging objects. Therefore, temporal contexts in DCVC-DC fail to predict the emerging objects well. By contrast, our SEVC utilizes an additional spatial feature, and the augmented spatial feature complements those regions. It can be observed that in the hybrid spatial-temporal contexts, those emerging objects are well described, thus providing a better prediction.

References

[1] VTM-17.0. https://vcgit.hhi.fraunhofer.de/

jvet/VVCSoftware_VTM. Accessed July 28, 2024. 1

[2] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999. 2

[3] 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 (ACM MM)*, pages 1503–1511, 2022. 1, 3

[4] Jiahao Li, Bin Li, and Yan Lu. Neural Video Compression with Diverse Contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22616–22626, 2023. 1, 2, 3

[5] Jiahao Li, Bin Li, and Yan Lu. Neural Video Compression with Feature Modulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26099–26108, 2024. 2, 3

[6] Zhuoyuan Li, Junqi Liao, Chuanbo Tang, Haotian Zhang, Yuqi Li, Yifan Bian, Xihua Sheng, Xinmin Feng, Yao Li, Changsheng Gao, et al. USTC-TD: A Test Dataset and Benchmark for Image and Video Coding in 2020s. *arXiv preprint arXiv:2409.08481*, 2024. 5

[7] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. SwinIR: Image Restoration Using Swin Transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1833–1844, 2021. 1, 2

[8] Xihua Sheng, Jiahao Li, Bin Li, Li Li, Dong Liu, and Yan Lu. Temporal Context Mining for Learned Video Compression. *IEEE Transactions on Multimedia*, 25:7311–7322, 2022. 1

[9] Shuwei Shi, Jinjin Gu, Liangbin Xie, Xintao Wang, Yujiu Yang, and Chao Dong. Rethinking Alignment in Video Super-Resolution Transformers. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:36081–36093, 2022. 1

[10] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1874–1883, 2016. 1

[11] Zachary Teed and Jia Deng. RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. In *European Conference on Computer Vision (ECCV)*, pages 402–419. Springer, 2020. 3

[12] 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. 5

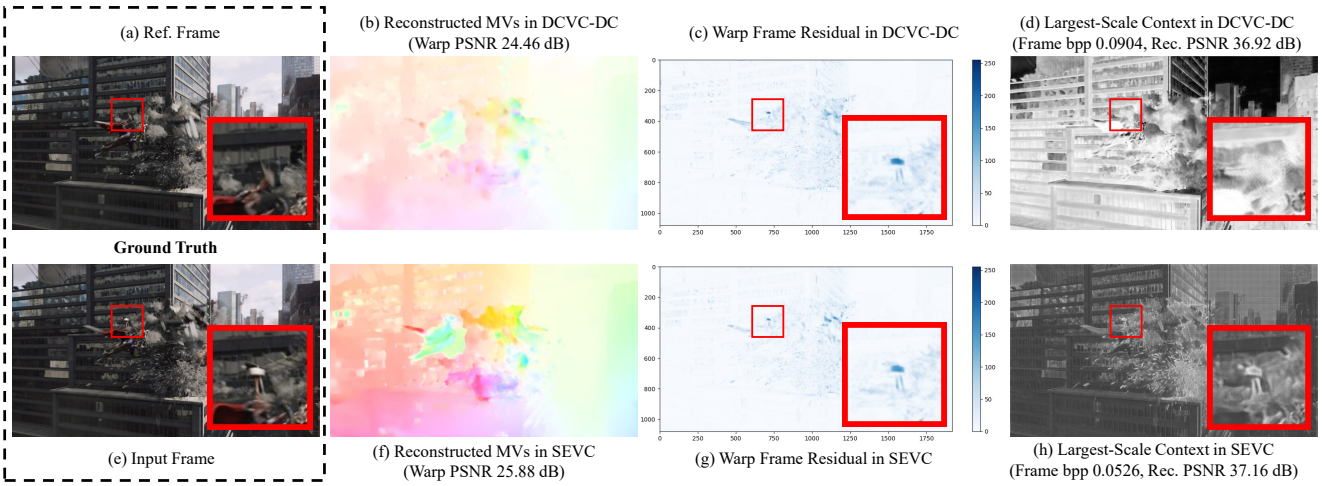

Figure g. Visualization of the MVs and contexts in DCVC-DC and our SEVC. This example is from *videoSRC22_1920x1080_24* video of MCL-JCV [12].

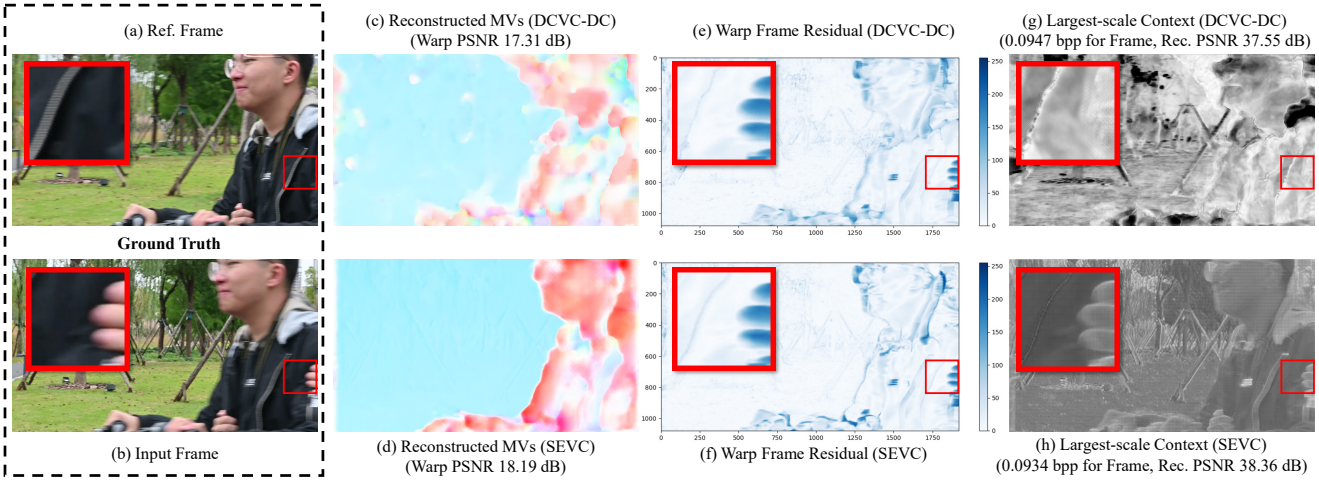

Figure h. Visualization of the MVs and contexts in DCVC-DC and our SEVC. This example is from *USTC_BycycleDriving* video of USTC-TD [6].

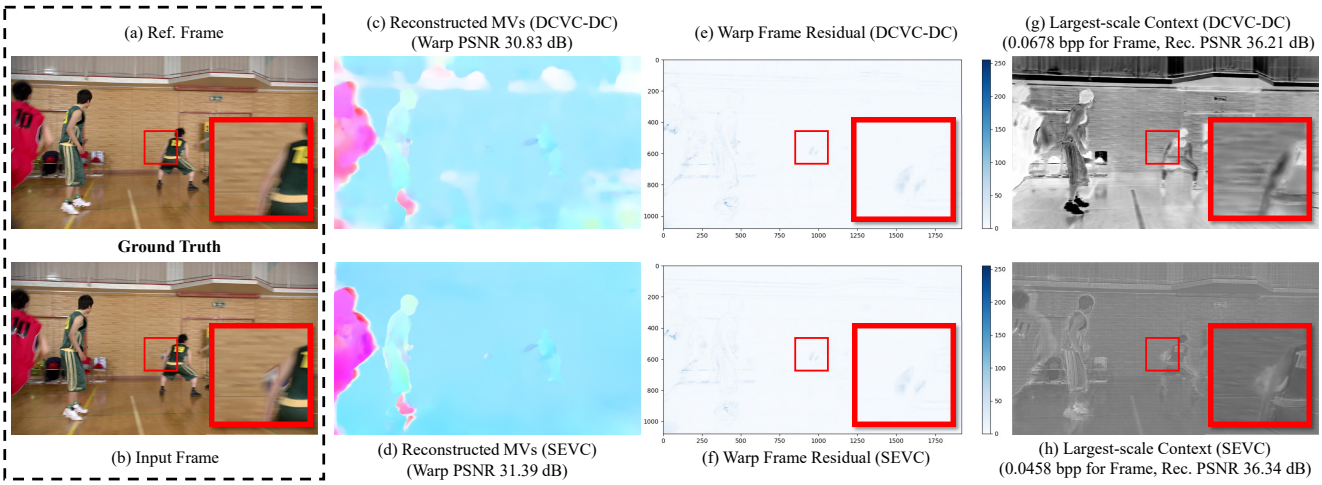

Figure i. Visualization of the MVs and contexts in DCVC-DC and our SEVC. This example is from *BasketballDrive_1920x1080_50* video of HEVC B [6].