MH-LVC: Multi-Hypothesis Temporal Prediction for Learned Conditional Residual Video Coding
Supplementary Materials

Huu-Tai Phung[1*] Zong-Lin Gao[1*] Yi-Chen Yao[1] Kuan-Wei Ho[1] Yi-Hsin Chen[1]
Yu-Hsiang Lin[1] Alessandro Gnutti[2] Wen-Hsiao Peng[1]

[1] National Yang Ming Chiao Tung University, Taiwan
[2] University of Brescia, Italy

This supplementary document provides the following additional materials and results to assist with the understanding of our MH-LVC:

- Implementation details in Section 2;
- Cascading and prediction errors in Section 3;
- Overview of Implicit Buffering Strategies in section 4
- Additional alternative Temporal Prediction Structures in section 5
- The number of long-term key frames in Section 6;
- Command lines for VTM and HM in Section 7;
- Comparison with the state-of-the-art methods in terms of MS-SSIM-RGB in Section 8;
- More visualizations in Section 9;

1. Mini-GOP

Table 1 examines the impact of the mini-GOP size on coding performance. These results justify our choice of mini-GOP 4.

Table 1. Ablation of different mini-GOP sizes. The anchor is LS with a mini-GOP size of 4.

mini-GOP Period	HEVC-B	UVG
mini-GOP 4	0.0	0.0
mini-GOP 8	1.0	0.5
mini-GOP 12	2.5	1.3
mini-GOP 16	2.9	2.1

2. Implementation Details

2.1. The Prediction Structure for Training

We adopt a 5-frame training strategy due to limited compute resources. We remark that our scheme can benefit from training on large GOPs and long sequences.

Fig. 1 (a) illustrates the temporal prediction structure during training. To create a quality structure among the decoded video frames, the weights w_t of $\{1.2, 0.5, 1.2, 0.9\}$ are assigned as follows: (1) 1.2 for Frame 2 (P^* frame), (2) 0.5 for Frame 3 with the quality level 2, (3) 1.2 for Frame 4 with the quality level 3, and (4) 0.9 for Frame 5 with the quality level 1. Notably, following a strategy similar to that of DCVC-DC [1], a specific feature extractor is trained to accommodate each of these weights. That is, in Fig. 2 (b) of the main paper, the feature extractor of the long-term key frame $\hat{x}_{key}$ changes with the quality level. The P^* frame right after the I-frame is unique in that it typically exhibits a much higher bitrate (and thus better decoded quality) than those of the remaining P-frames due to error propagation aware training [2]. In the setting with an infinite intra period, we enable P^* frames periodically to mitigate temporal cascading errors.

To be as consistent with the prediction scenario at inference time as possible, we always use the most recently decoded frame as the short-term reference frame during training. Likewise, the I-frame, due to its higher quality, is always used as the long-term key frame. However, at inference time, the long-term key frame can be an I-frame, a P^* frame, or a P-frame with the quality level 3.

2.2. The Prediction Structures for Inference

Fig. 1 (b) illustrates our prediction structure at inference time and how the decoded frame buffer evolves over time under an intra period of 32. For forming a 4-frame mini-GOP, we follow a quality pattern similar to the hierarchical P prediction in traditional codecs. For the first mini-GOP where no prior key frames are available, the I-frame and P^* frame are stored in the long-term section to serve as the long-term reference frames for the subsequent frames.

Fig. 1 (c) illustrates the case with an infinite intra period, where we enable P^* frames periodically to mitigate tempo-

*Equal contribution.

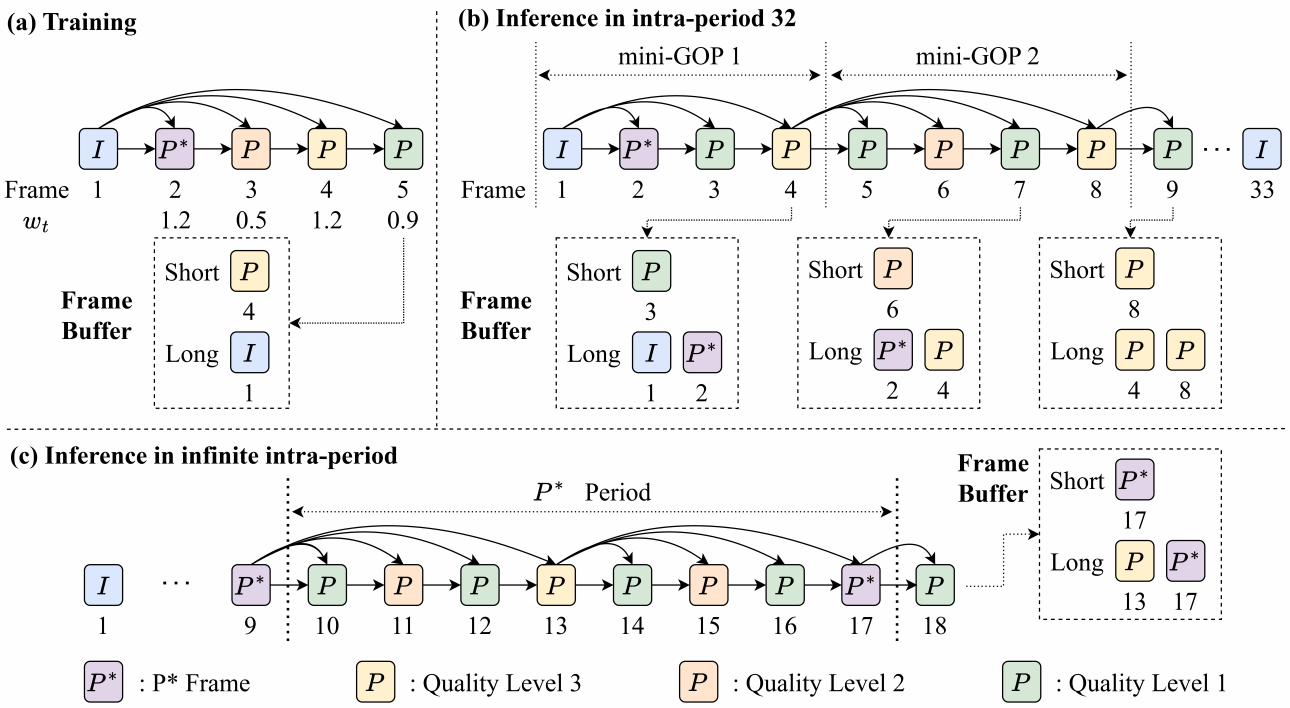

Figure 1. Illustration of the coding structures for (a) training with intra-period 32, (b) inference with intra-period 32, and (c) inference with an infinite intra-period.

Table 2. Training procedure. MENet, MWNet, MCNet represent the motion estimation network, the motion extrapolation network, and the motion compensation network, respectively. R^{motion} and R_t represent the motion and total bitrates, respectively. EPA is error propagation aware training.

Phase	Number of Frames	Training Modules	Loss	lr	Epoch
ME Training	2	MENet	$D(x_t, \text{warp}(x_{t-1}, f_t))$	1e-5	5
Motion Coding	3	MWNet & Motion codec	$R^{\text{motion}} + \lambda \times D(x_t, \text{warp}(x_{t-1}, \hat{f}_t))$	1e-4	10
MC Training	3	MCNet	$R^{\text{motion}} + \lambda \times D(x_t, \hat{x}_C)$	1e-4	3
Inter-Frame Coding	3	Inter codec	$R_t + \lambda \times D(x_t, \hat{x}_t)$	1e-4	10
	5	Inter codec	$R_t + \lambda \times D(x_t, \hat{x}_t)$	1e-4	2
	5	Inter codec & MCNet	$R_t + \lambda \times D(x_t, \hat{x}_t)$	1e-4	7
Finetune	5	All modules except MENet	$R_t + \lambda \times D(x_t, \hat{x}_t)$	1e-4	5
Finetune + EPA	5	All modules except MENet	$R_t + \lambda \times D(x_t, \hat{x}_t)$	1e-5	1
	5	All modules except MENet	$R_t + \lambda \times w_t \times D(x_t, \hat{x}_t)$	1e-5	2
Variable Rate Finetune	5	All modules except MENet	$R_t + \lambda_p \times w_t \times D(x_t, \hat{x}_t)$	1e-5	3

ral cascading errors.

2.3. Training Procedures

Table 2 summarizes our training procedure. It begins with training the single-rate model, followed by fine-tuning it to arrive at the variable-rate model. The code will be made available for reproducibility upon the acceptance of the paper.

3. Cascading and Prediction Errors

Fig. 3 shows that our LS is more effective than TP in mitigating temporal cascading errors. The results stress the importance of incorporating both long- and short-term refer-

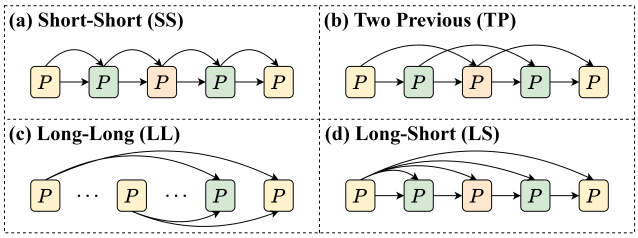

Figure 2. Alternative prediction structures.

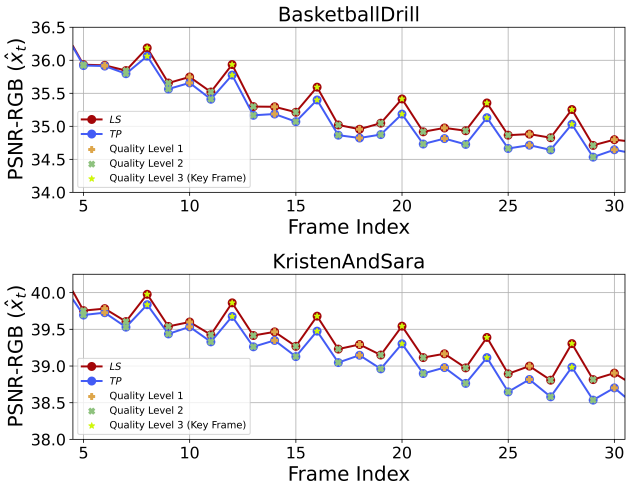

Figure 3. The per-frame PSNR profiles on BasketballDrill and KristenAndSara: LS versus TP. Their average bitrates are comparable.

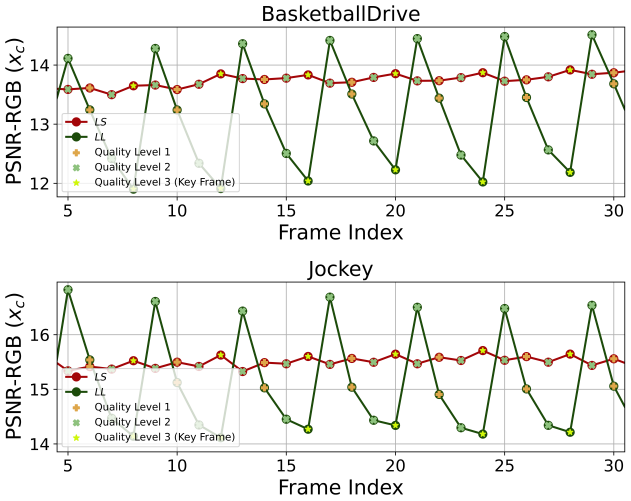

Figure 4. The per-frame PSNR profiles of the temporal predictor x_c on BasketballDrive and Jockey: LS versus LL. Their average bitrates are comparable.

ence frames.

Fig. 4 further evaluates the prediction errors of LS and LL by visualizing the PSNR-RGB of the temporal predictor x_c. These results are evaluated for fast-motion sequences, where the prediction errors are more noticeable. LL exhibits inferior temporal predictor quality compared to LS. Notably, the quality of the temporal predictor with LL degrades over time within a mini-GOP due to the increasing prediction distance between the coding frame and its reference frames. In contrast, the temporal predictor quality of LS remains relatively stable over time, as it leverages both long- and short-term reference frames.

4. Overview of Implicit Buffering Strategies.

Figure 5 provides an overview of the implicit buffering strategies framework. To ensure a fair comparison, we implement the implicit buffering strategy of DCVC-TCM [3] within the same conditional residual video coding framework as our proposed MH-CRT (see Section 4.3 in the main paper).

5. Additional Alternative Temporal Prediction Structures

Table 3. BD-rate comparison of several prediction structures with LS^+ serving as the anchor.

	LS	LS^+	SS	TP	TP^+
HEVC-B	0.0	**-3.4**	14.6	3.3	-0.8
UVG	0.0	**-2.7**	10.5	2.4	-0.5
MCL-JCV	0.0	**-4.6**	10.5	2.4	-3.2

We further investigate several alternative prediction structures (see Fig. 2) to prove the effectiveness of LS. As mentioned in the main paper, (i) "Short-Short (SS), which simulates the effect of the single-hypothesis prediction with our two-hypothesis framework by referencing the most recently decoded frame twice yet with the same optical flow map, (ii) "Two Previous (TP)," which predicts a current frame from the last two previously decoded frames. We also include "Two Previous Plus (TP^+)," which uses the most recently decoded frame $\hat{x}_{t-1}$ as the short-term reference frame and selects adaptively another short-term reference frame from $\hat{x}_{t-2}$, $\hat{x}_{t-3}$, (iv) "Long-Long (LL)," which predicts from the last two long-term key frames.The "Long-Short (LS)" corresponds to our MH-LVC-1 prediction scheme, which has one short-term reference frame and one long-term key frame. Following the same notation as TP^+, we use LS^+ to denote our MH-LVC-2, which has two long-term key frames for adaptive prediction. Note that all the prediction structures share the same network weights trained solely for the LS prediction. Table 3 justifies the effectiveness of LS^+ over SS, TP and TP^+.

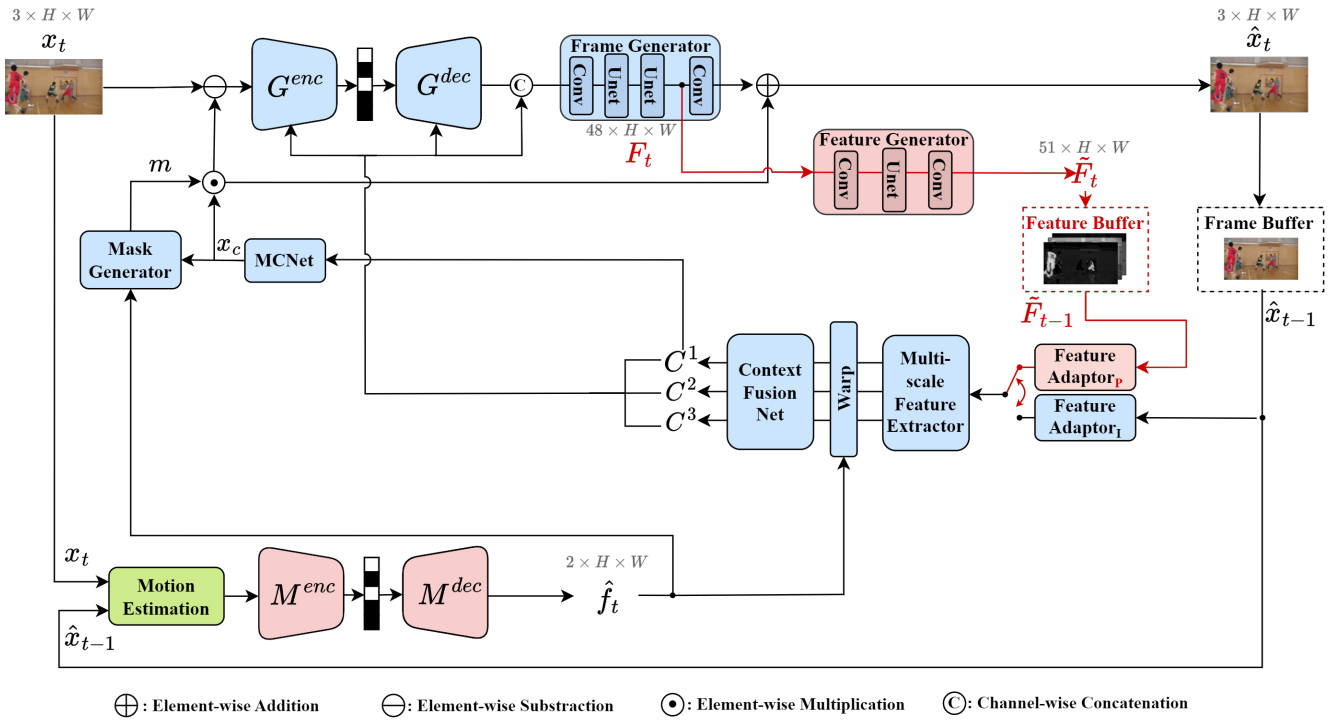

Figure 5. Overview of the implicit buffering strategies framework.

Table 4. Ablation on the number of key frames for online long-term key frame selection. The anchor is MH-LVC-1.

Number of Key Frames	Encoding KMACs/pixel	BD-rate (%) PSNR-RGB	
		HEVC-B	UVG
1	1507	0.0	0.0
2	2611	-3.4	-2.7
3	3716	-3.8	-3.0

6. The Number of Long-term Key Frames

Table 4 presents how the number of long-term key frames under our LS prediction structure may impact the complexity and compression performance. It is seen that the coding gain diminishes when the number of long-term key frames goes beyond 2, while the encoding kMAC/pixel increases considerably.

7. Command Lines for VTM and HM

We compare MH-LVC with traditional video codecs, including VTM-17.0 and HM-16.25. Following [1], we have these codecs encode input videos in YUV444 format (by converting them from YUV420 into YUV444). The reconstructed YUV444 videos are then transformed into RGB domain for evaluating the distortions. For HM and VTM, *encoder_lowdelay_main_rext.cfg* and *encoder_lowdelay_vtm.cfg* config files are used, respectively. The command lines used are as follows:

- `-c {config file name}`
  `--InputFile={input video name}`
  `--InputBitDepth={input bit depth}`
  `--OutputBitDepth={output bit depth}`
  `--InputChromaFormat=444`
  `--FrameRate={frame rate}`
  `--DecodingRefreshType={refresh type}`
  `--FramesToBeEncoded=96`
  `--SourceWidth={width}`
  `--SourceHeight={height}`
  `--IntraPeriod={intra period}`
  `--QP={qp}`
  `--BitstreamFile={bitstream file name}`

where we set *DecodingRefreshType* and *IntraPeriod* to 2 and 32, respectively, for an intra period of 32. They are set to 0 and -1, respectively, for an infinite intra period.

8. Comparison with the State-of-the-art Methods in Terms of MS-SSIM-RGB

Fig. 6 and Fig. 7 show the rate-distortion curves of our MH-LVC under an intra-period of 32 and infinity, respectively, where the quality metric is MS-SSIM-RGB.

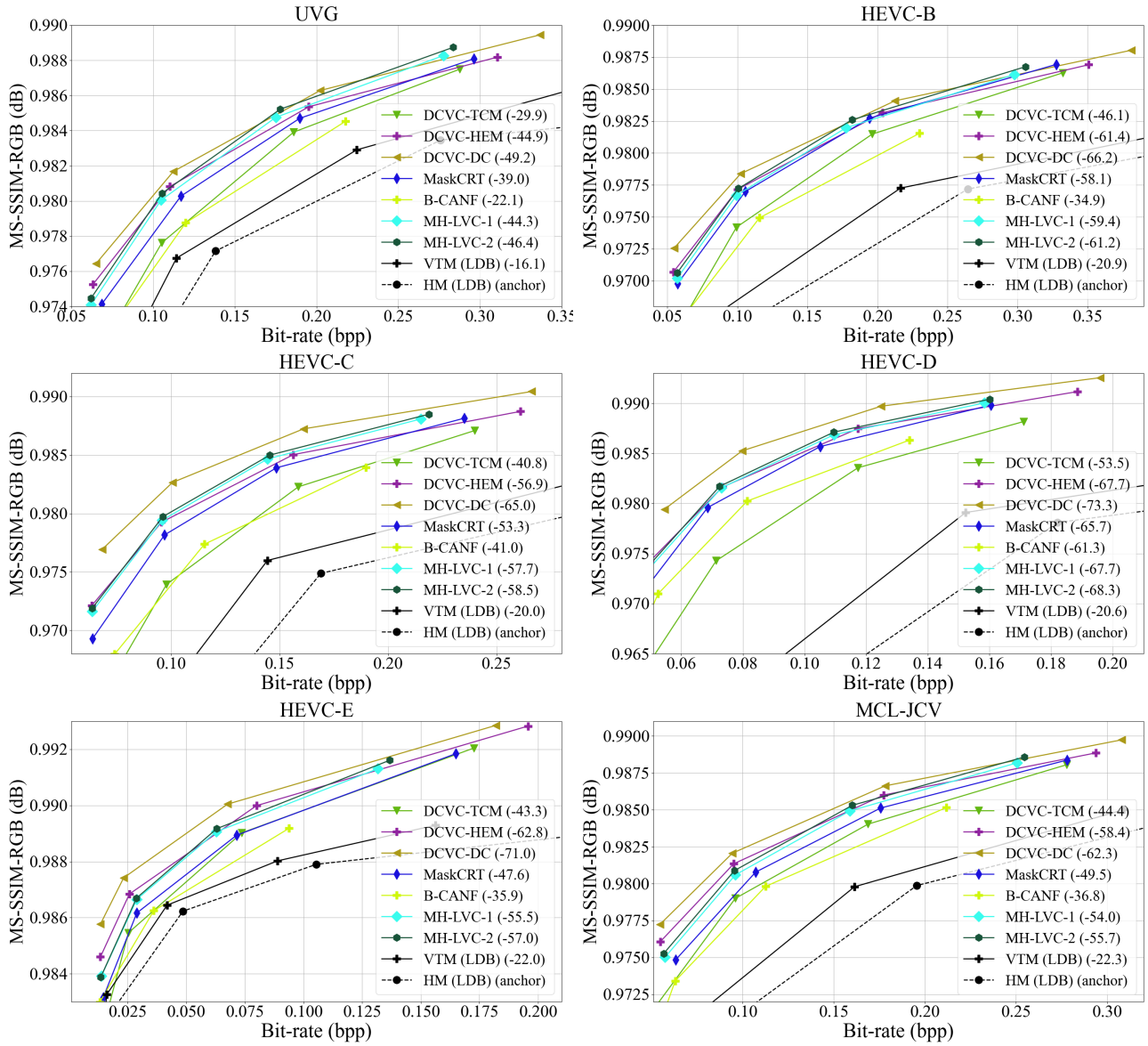

Figure 6. Rate-distortion performance comparison with intra-period 32 in terms of MS-SSIM-RGB. The numbers within the parentheses are BD-rates, with HM-16.25 (Low-delay B) serving as the anchor.

9. More Visualizations

Fig. 10 presents more visualizations for the gating signal generated by the spatial gate predictor.

References

[1] Jiahao Li, Bin Li, and Yan Lu. Neural video compression with diverse contexts. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, Canada, June 18-22, 2023*, 2023. 1, 4

[2] Guo Lu, Chunlei Cai, Xiaoyun Zhang, Li Chen, Wanli Ouyang, Dong Xu, and Zhiyong Gao. Content adaptive and error propagation aware deep video compression. In *European Conference on Computer Vision (ECCV)*, pages 456–472. Springer, 2020. 1

[3] Xihua Sheng, Jiahao Li, Bin Li, Li Li, Dong Liu, and Yan Lu. Temporal context mining for learned video compression. *IEEE Transactions on Multimedia*, pages 1–12, 2022. 3

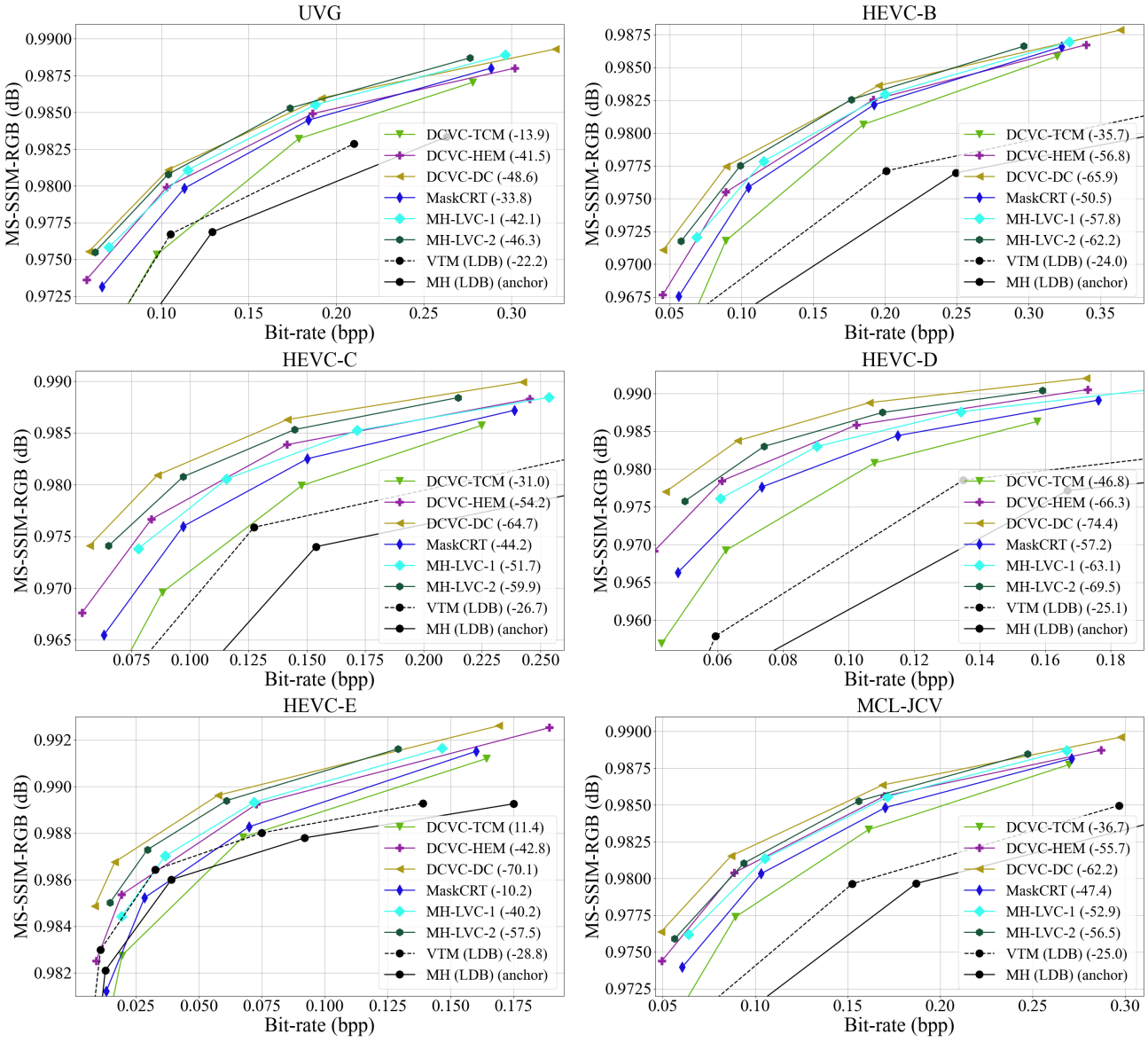

Figure 7. Rate-distortion performance comparison under an infinite intra-period in terms of MS-SSIM-RGB. The numbers within the parentheses are BD-rates, with HM-16.25 (Low-delay B) serving as the anchor.

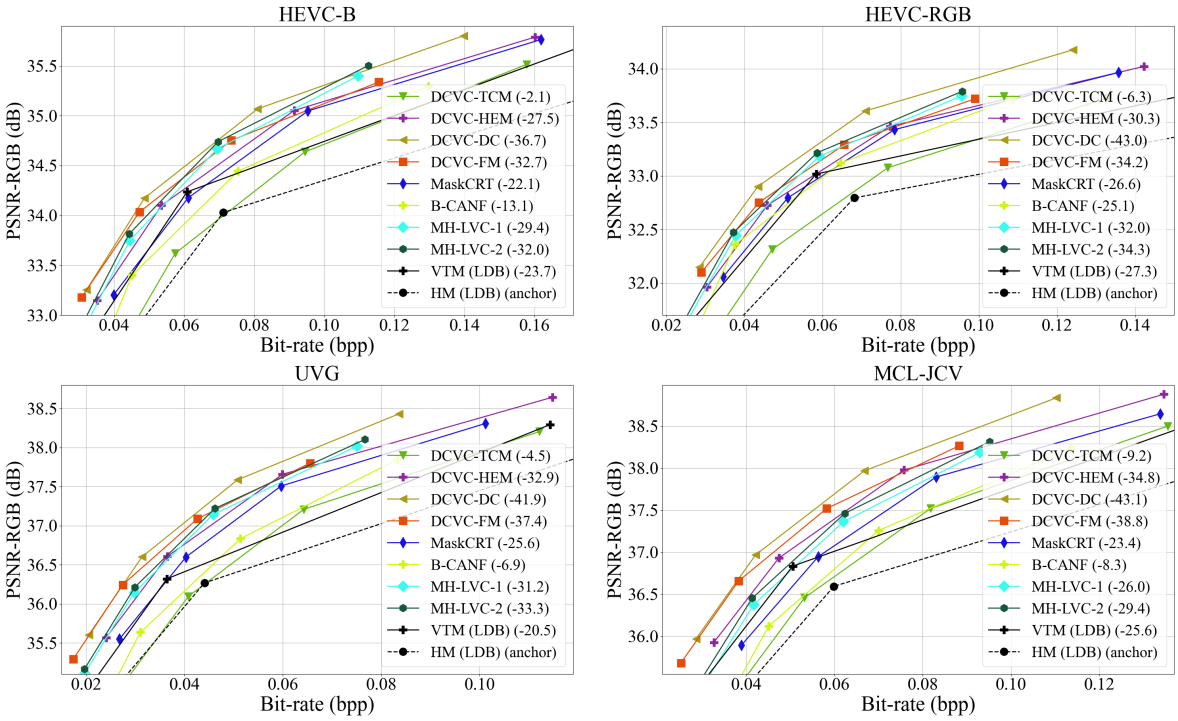

Figure 8. Rate-distortion performance comparison with intra-period 32 in terms of PSNR-RGB. The numbers within the parentheses are BD-rates, with HM-16.25 (Low-delay B) serving as the anchor.

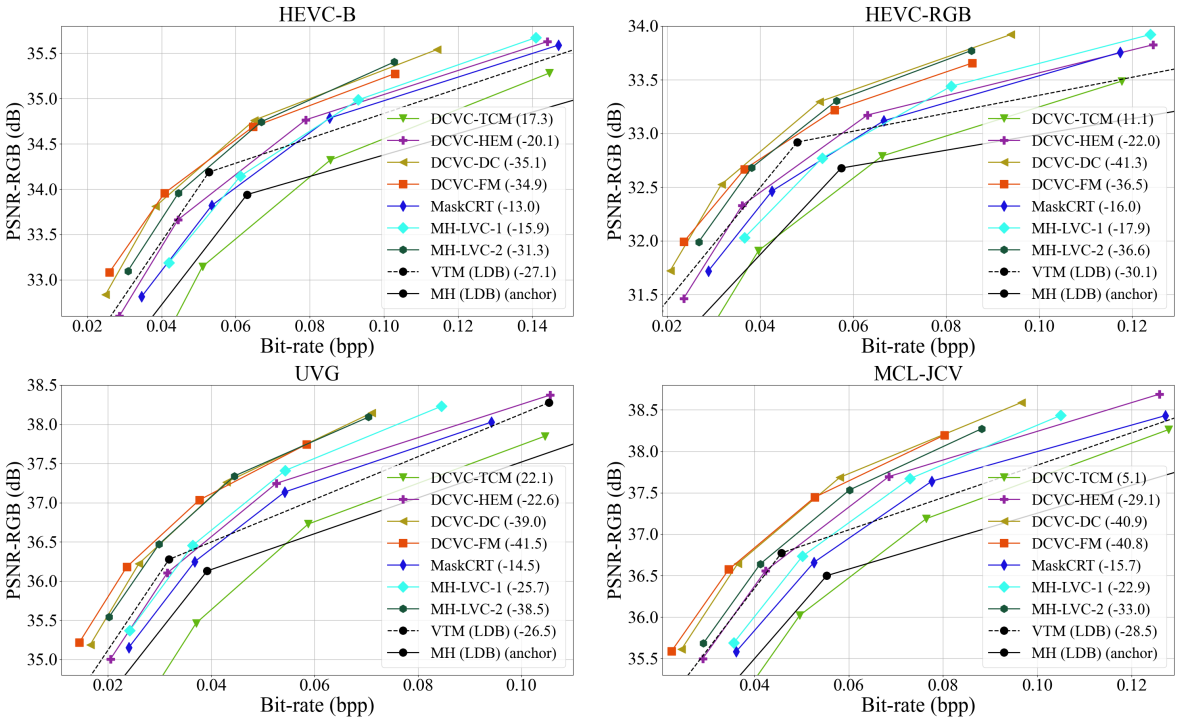

Figure 9. Rate-distortion performance comparison under an infinite intra-period in terms of PSNR-RGB. The numbers within the parentheses are BD-rates, with HM-16.25 (Low-delay B) serving as the anchor.

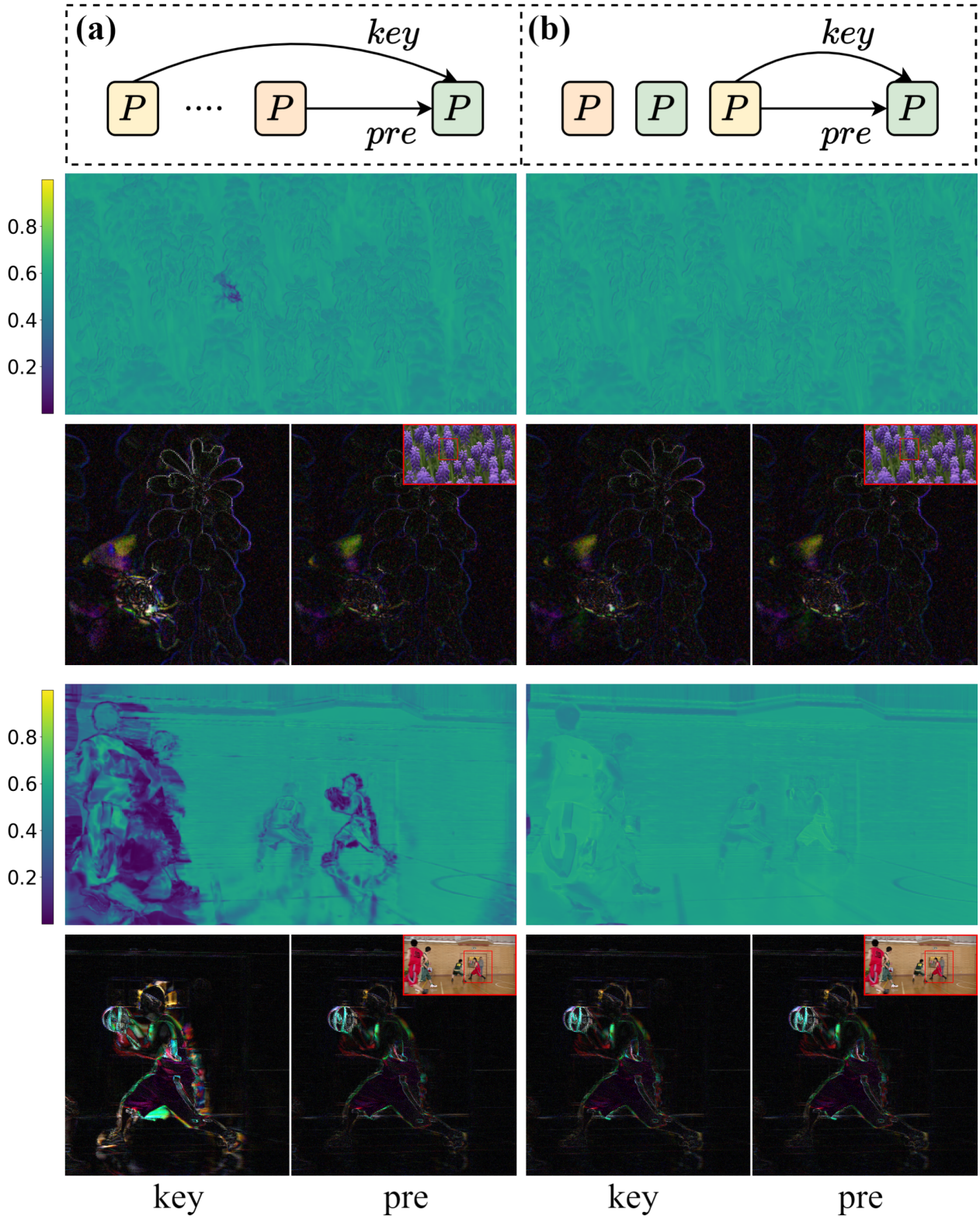

Figure 10. Visualization of the gating signal $\gamma^{(1)}$ for two temporal prediction structures. (a) adopts both long- and short-term reference frames, (b) has two predictors derived from the same short-term reference frame with the same optical flow map. The bottom row displays the prediction residues between the coding frame x_t and its two motion-compensated reference frames $\hat{x}_{key}$ (denoted as key) and $\hat{x}_{t-1}$ (denoted as pre).