Generative Neural Video Compression via Video Diffusion Prior

Supplementary Material

A. Summary

This supplementary material provides additional implementation details, extended experiments, more visual results, and further discussion. It is organized as follows:

- Section B describes the experimental setup, including test sequence preprocessing and the reproduction details of all baseline methods.
- Section C provides the full implementation details of the Contextual Latent Codec module.
- Section D presents additional results, including distortion metrics (PSNR, MS-SSIM), perceptual metrics (LPIPS-Alex), analyses of temporal consistency and semantic continuity (E_{warp}, CLIP-F, and FVD), complexity evaluation, user study results, and more visual examples.
- Section E provides a discussion of GNVC-VD in comparison with GLVC.
- Section F discusses the current limitations of GNVC-VD and outlines several directions for future work.

B. Test Settings

For fair comparison with both traditional codecs and neural video compression methods, all approaches are evaluated in the RGB color space.

B.1. Test Sequences

The raw videos are stored in YUV420 format. We convert them to RGB using the BT.709 standard. For evaluation, we extract the first 96 frames of each sequence. For codecs that require input resolutions to be multiples of 64, we apply zero-padding before encoding and crop the decoded frames back to their original size.

B.2. Traditional Codecs

We evaluate two representative traditional codecs, HM-16.25[1] and VTM-17.0[2]. Both operate internally in 10-bit YUV444, and final results are computed in RGB. We use the official low-delay configurations *encoder_lowdelay_rext.cfg* (HM) and *encoder_lowdelay_vtm.cfg* (VTM).

B.3. Neural-based Codecs

Implementation details for neural codecs are summarized below:

- **DCVC-FM / DCVC-RT**. We use the official code and checkpoints from the authors' GitHub repository[3] . All frames are processed in RGB, and the GOP size is set to 96.
- **GLC-Video**. We use the reconstructed videos and bitrates provided directly by the original authors of GLC-Video [10]. All evaluation metrics are computed from the provided reconstructions.
- **PLVC**. PLVC [13] is evaluated using its official implementation[4] and pre-trained weights. Since PLVC adopts HiFiC [9] for I-frame coding, we use its PyTorch implementation[5] for consistency.
- **GNVC-VD**. Due to training and inference constraints, GNVC-VD processes each 96-frame sequence as four GOPs with lengths of 25, 25, 25, and 21 frames, respectively.

C. Model Implementation Details

Fig. 1 illustrates the detailed architecture of the proposed Contextual Latent Codec module. We use two separate neural networks to perform transform coding on the anchor latent l_1 and the predictive latents $l_{t t>1}$.

Anchor latent (I-frame). The processing pipeline for the anchor latent is shown in Fig. 1(a). We adopt a design similar to ELIC [4], where the analysis and synthesis transforms (g_s and g_a) are constructed from cascaded residual bottleneck blocks [5] and attention blocks [1]. A joint space–channel context model estimates the probability distribution of the quantized anchor latent $\hat{y}_1$.

Predictive latents (P-frames). As illustrated in Fig. 1(b), for the predictive latents, we follow the architecture of DCVC-RT [6], where the transforms g_s and g_a are built from cascaded DC Blocks [6]. To balance coding efficiency and reconstruction quality, we adopt the two-step distribution estimation scheme described in [8].

D. Additional Experiments

D.1. Additional Metrics Evaluation

For a more comprehensive comparison, we report the rate–distortion curves of all baseline methods and our GNVC-VD in terms of PSNR, MS-SSIM, and LPIPS-Alex in Fig. 2. The VGG-based LPIPS variant correlates more strongly with human perception in generative compression.

[1] https://vcgit.hhi.fraunhofer.de/jvet/HM

[2] https : / / vcgit . hhi . fraunhofer . de / jvet / VVCSoftware_VTM

[3] https://github.com/microsoft/DCVC

[4] https://github.com/RenYang-home/PLVC

[5] https://github.com/Justin-Tan/high-fidelity-generative-copression

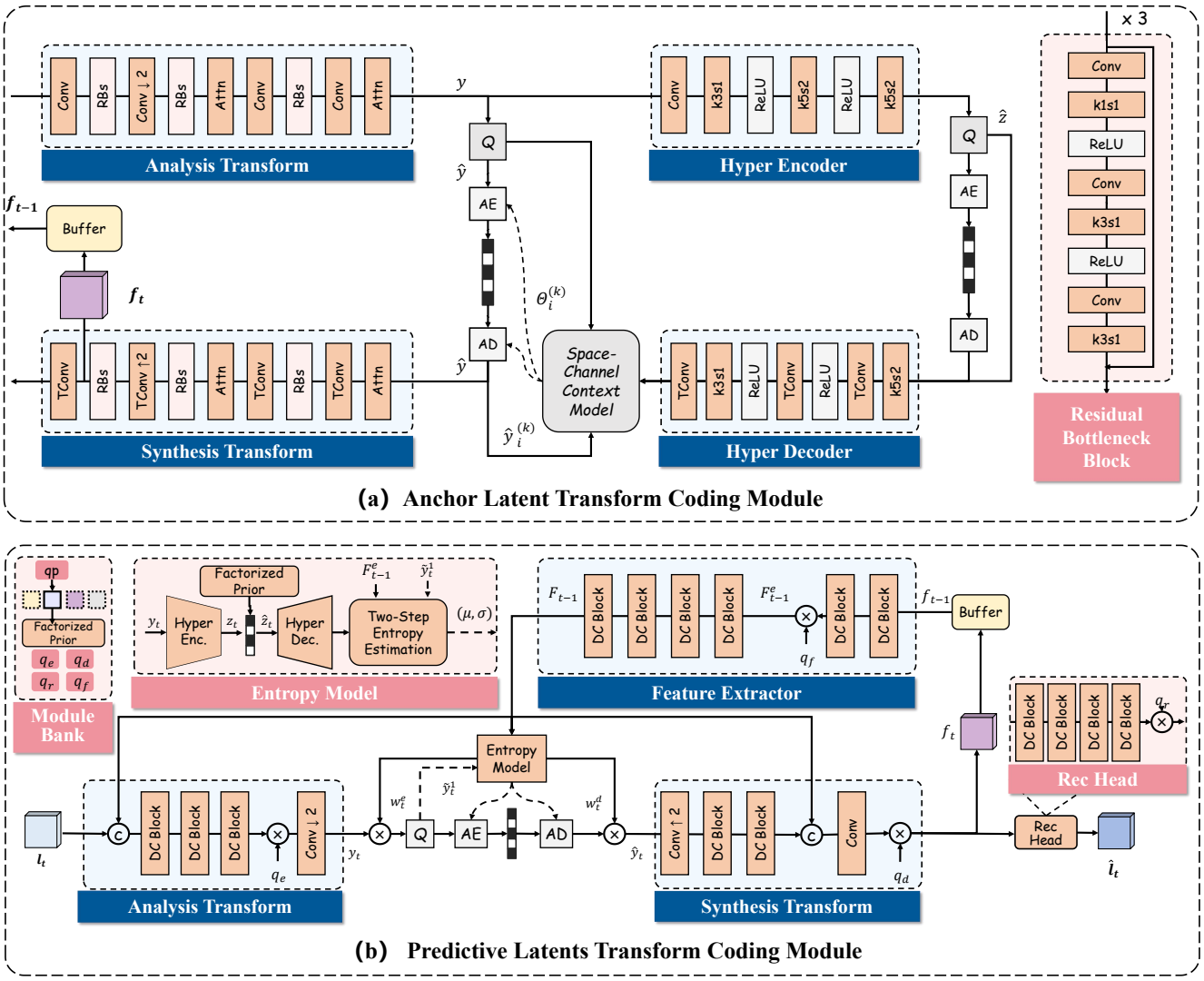

Figure 1. **Architecture of the Contextual Latent Codec module.**

Therefore, in the main paper, perceptual comparisons are reported using LPIPS-VGG, which provides a more reliable indicator of perceptual fidelity. However, because the AlexNet-based LPIPS metric is more commonly used in the learned compression literature, we additionally include LPIPS-Alex results here for completeness. Compared with perceptual codecs such as GLC-Video [10] and PLVC [13], GNVC-VD achieves clear improvements in distortion-oriented metrics (PSNR and MS-SSIM) while also delivering notably better perceptual quality (LPIPS-Alex), consistent with the LPIPS-VGG and DISTS improvements reported in the main paper. Relative to MSE-optimized codecs, although a small gap remains in PSNR and MS-SSIM, GNVC-VD provides substantially superior perceptual fidelity.

D.2. Additional Analysis on Temporal Consistency

Table 1 reports the per-sequence results of bitrate, E_{warp} [7], and CLIP-F [11]. Fig. 3 shows the rate–temporal-consistency curves of GNVC-VD, DCVC-RT [6], and GLC-Video [10] on HEVC Class B [2], measured by FVD [12]. As a generative codec built on an image-domain prior, GLC-Video exhibits relatively weak temporal consistency on most sequences. In contrast, GNVC-VD, which leverages a video-domain prior, achieves markedly stronger temporal coherence. Moreover, benefiting from its superior perceptual quality, GNVC-VD also achieves better FVD than DCVC-RT. Although GNVC-VD yields slightly lower semantic consistency than traditional and MSE-optimized codecs, it still substantially outperforms the image-prior-based generative codec GLC-Video.

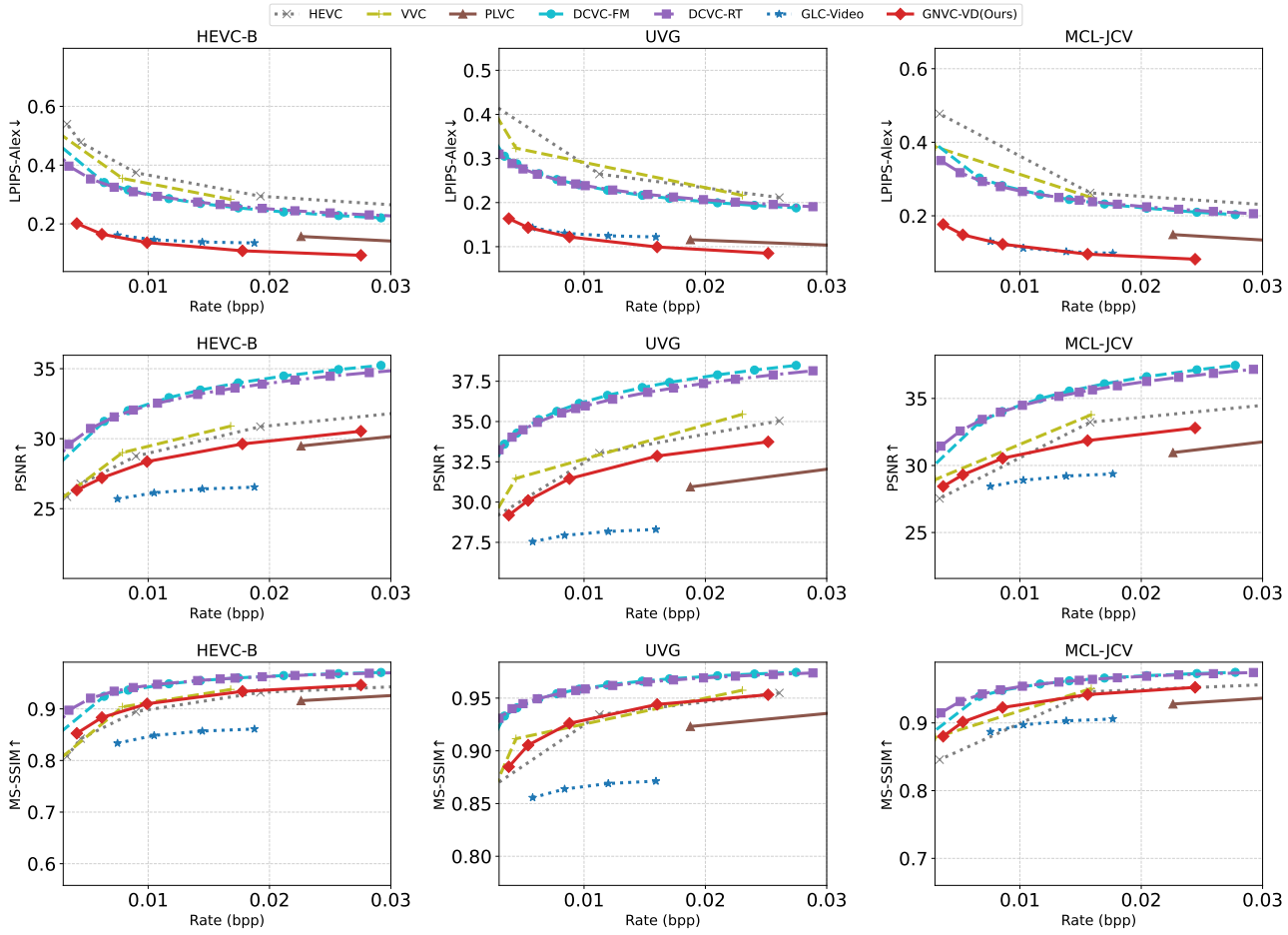

Figure 2. **Rate–distortion curves of all codecs evaluated using LPIPS-Alex, PSNR, and MS-SSIM.**

Table 1. **Detailed bpp, E_{warp}, and CLIP-F results for all codecs on HEVC-B.**

Video name	HEVC			VVC			DCVC-FM			DCVC-RT			GLC-Video			GNVC-VD		
	bpp	$E_{\text{warp}}\downarrow$	CLIP-F↑	bpp	$E_{\text{warp}}\downarrow$	CLIP-F↑	bpp	$E_{\text{warp}}\downarrow$	CLIP-F↑	bpp	$E_{\text{warp}}\downarrow$	CLIP-F↑	bpp	$E_{\text{warp}}\downarrow$	CLIP-F↑	bpp	$E_{\text{warp}}\downarrow$	CLIP-F↑
BasketballDrive	0.0098	61.14	0.974	0.0087	65.34	0.976	0.0069	259.44	0.972	0.0052	255.84	0.972	0.0089	263.77	0.967	0.0057	263.33	0.967
BQTerrace	0.0073	7.98	0.993	0.0066	7.89	0.991	0.0060	5.61	0.993	0.0050	5.61	0.992	0.0066	41.47	0.984	0.0065	15.69	0.991
Cactus	0.0098	15.06	0.972	0.0077	15.27	0.973	0.0060	9.35	0.976	0.0051	9.12	0.976	0.0064	27.45	0.973	0.0061	14.96	0.979
Kimono1	0.0091	24.18	0.986	0.0082	25.45	0.988	0.0066	17.42	0.989	0.0051	17.78	0.991	0.0097	40.30	0.987	0.0060	20.89	0.988
ParkScene	0.0087	8.07	0.990	0.0081	8.03	0.991	0.0063	7.24	0.989	0.0057	7.45	0.990	0.0056	59.88	0.983	0.0064	18.48	0.986
average result	0.0089	23.29	0.982	0.0079	24.40	0.984	0.0064	59.81	0.984	0.0052	59.16	0.984	0.0074	86.57	0.979	0.0061	66.67	0.982

D.3. Complexity

Model Parameter Count. As summarized in Table 2, the proposed GNVC-VD contains 2334.5M parameters in total, including 126.9M in the 3D VAE, 53.1M in the contextual latent codec module, and 2154.5M in the VideoDiT.

Coding Speed at Different Resolutions. Table 3 reports the per-frame encoding and decoding latency on a single A800 GPU. At 1920×1080, GNVC-VD requires 153 ms/frame for encoding and 1557 ms/frame for decoding. The latency decreases to 58/386 ms/frame at 1080×720 and further to 25/129 ms/frame at 640×480.

Computational Complexity. As shown in Table 4, for a 25-frame video clip at 1080p resolution, the peak memory usage reaches 71.41 GB, while the encoder and decoder require 121 kMACs/pixel and 10954 kMACs/pixel, respectively. At 720p, the peak memory usage decreases to 40.9 GB, with encoding and decoding complexity of 129 kMACs/pixel and 11702 kMACs/pixel, respectively. At 480p, the peak memory usage is 29.36 GB, and the corresponding encoding and decoding complexity is 129 kMACs/pixel and 11661 kMACs/pixel, respectively.

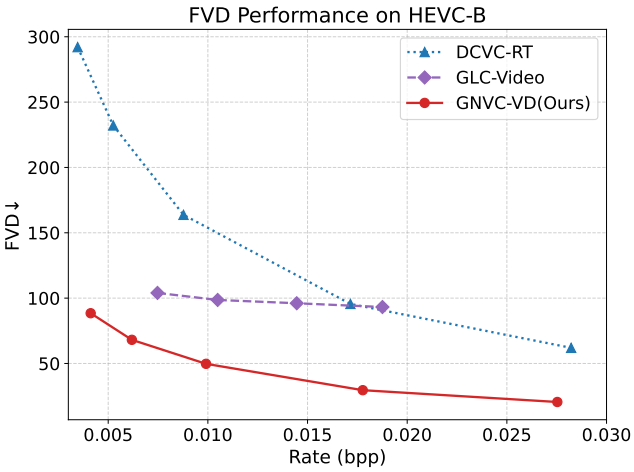

Figure 3. **Rate–FVD comparison on HEVC Class B.** Lower FVD indicates better temporal consistency and perceptual realism. GNVC-VD consistently achieves the lowest FVD across the bitrate range, outperforming both DCVC-RT and GLC-Video.

Table 2. **Parameter count of each major module in the proposed GNVC-VD framework.**

Module Name	**Parameters (M)**
3D VAE	126.9
Contextual Latent Codec	53.1
VideoDiT	2154.5
Total	**2334.5**

Table 3. **Coding speed with different resolutions on a single A800 GPU.**

Resolutions	1920 × 1080	1080 × 720	640 × 480
Encoding	153 ms	58 ms	25 ms
Decoding	1557 ms	386 ms	129 ms

Table 4. **Computational complexity of GNVC-VD.**

Resolutions	Enc./Dec. kMACs	Peak Mem. (GB)
1920 × 1080	121/10954	71.41
1080 × 720	129/11702	40.90
640 × 480	129/11661	29.36

D.4. User Study

To assess perceptual quality and temporal stability, we conducted a user study comparing GNVC-VD with VVC, DCVC-RT, DCVC-FM, and GLC-Video. In each trial, participants viewed the reference video at the top and two reconstructed versions below it—one produced by GNVC-VD and the other by a baseline codec. The left–right order

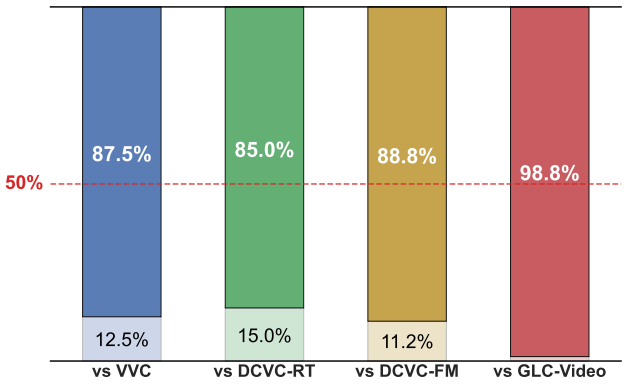

Figure 4. **User study results comparing GNVC-VD against VVC, DCVC-RT, DCVC-FM, and GLC-Video.** The bars show the percentage of participants who preferred GNVC-VD in pairwise comparisons.

was randomized to avoid positional bias. Participants were instructed to select the reconstruction that better matched the reference in terms of perceptual quality and temporal stability. As illustrated in Fig. 4, across all pairwise comparisons, GNVC-VD received strong user preference, achieving over **85%** preference against both traditional and neural codecs, and nearly unanimous preference against the image-prior-based GLC-Video. These subjective findings are consistent with the objective evaluations, providing a complementary assessment of GNVC-VD's perceptual fidelity and temporal coherence.

D.5. Additional Visual Examples

We provide additional qualitative comparisons on three datasets: HEVC Class B, MCL-JCV, and UVG. As shown in Fig. 5, GNVC-VD consistently outperforms prior state-of-the-art methods, delivering higher visual fidelity across diverse content while operating at the lowest bitrate.

E. Discussion of GLVC

GLVC [3] mainly focuses on the *spatio-temporal latent compression* of 3D-VAE latent representations. In contrast, GNVC-VD addresses not only the compression of spatio-temporal latents but also *sequence-level latent refinement* via a **VideoDiT**-based prior with explicit temporal coupling. This design is particularly important for mitigating temporal flickering and improving temporal consistency across video frames.

F. Limitation and Future Work

Although GNVC-VD achieves strong perceptual quality and temporal consistency at ultra-low bitrates, several limitations remain. First, the efficiency of the transform coding module can be further improved to enhance overall coding performance. Second, diffusion-based latent refinement

still incurs non-negligible computational overhead, making acceleration an important direction for future work.

Another limitation lies in variable-bitrate modeling. Unlike distortion-oriented variable-rate codecs, which mainly vary quantization strength, GNVC-VD couples rate control with sequence-level generative refinement. A unified variable-rate design would therefore require joint rate conditioning of both the transform codec and the VideoDiT prior during flow matching. In our preliminary experiments, this design resulted in unstable optimization and degraded temporal consistency. For this reason, the current framework adopts fixed-rate models to ensure stable training and reliable performance in the ultra-low-bitrate regime, while unified variable-rate modeling remains an important topic for future work.

Finally, although GNVC-VD supports long videos through chunk-wise decoding, practical streaming deployment would require causal temporal attention and a rolling latent buffer, such as a sliding-window mechanism. Improving the efficiency of long-sequence decoding, potentially via model distillation, is another important direction for future research.

Overall, addressing these limitations would further improve the practicality and scalability of GNVC-VD in real-world ultra-low-bitrate video compression.

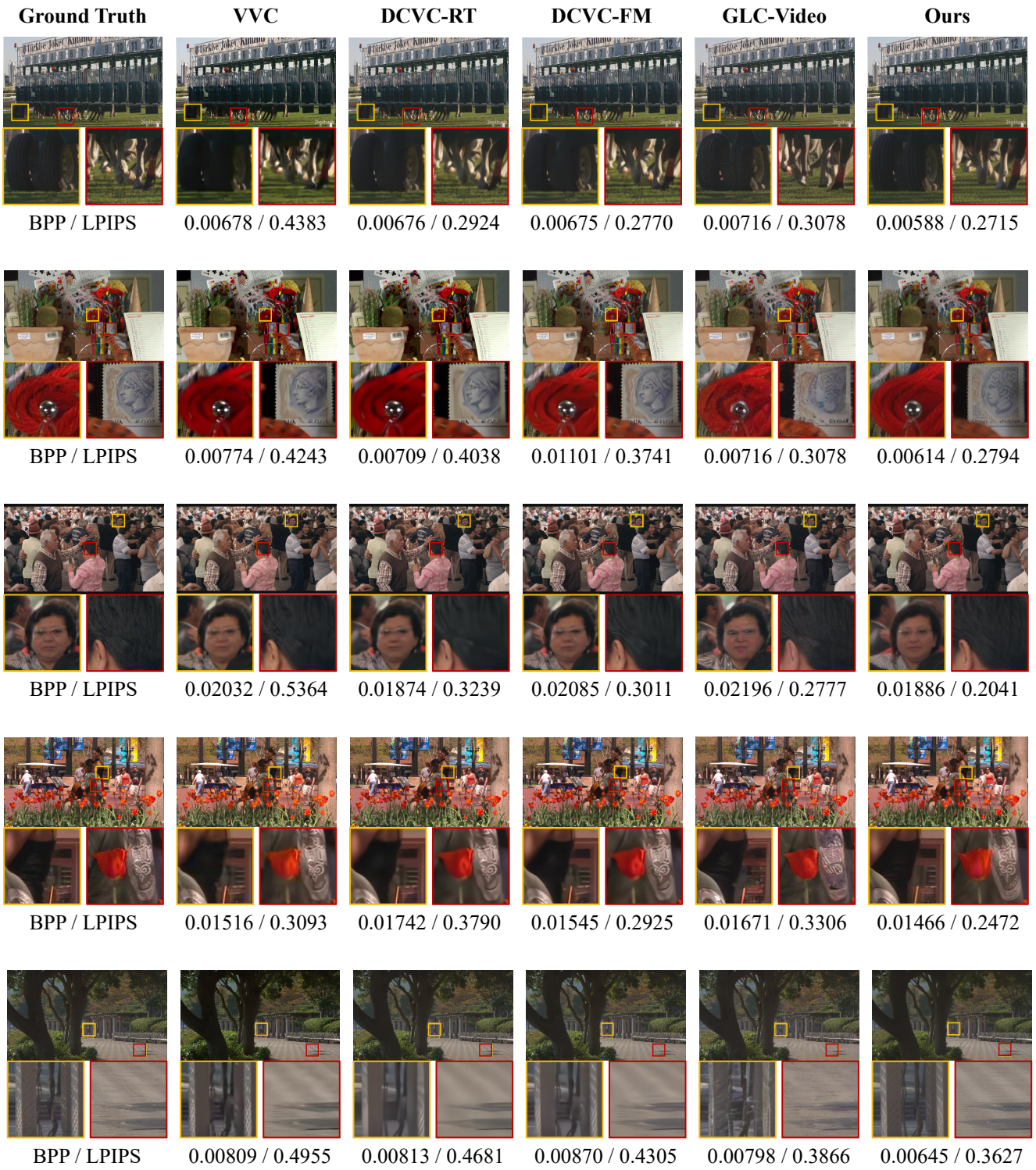

Figure 5. **Visual comparisons across several test sequences, including ground truth, VVC, DCVC-RT, DCVC-FM, GLC-Video, and our GNVC-VD.** Zoomed-in patches highlight texture preservation and perceptual differences. Bitrate (bpp) and LPIPS scores are shown beneath each reconstruction.

References

[1] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7939–7948, 2020. 1

[2] 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. 2

[3] Zongyu Guo, Zhaoyang Jia, Jiahao Li, Xiaoyi Zhang, Bin Li, and Yan Lu. Generative latent video compression. *arXiv preprint arXiv:2510.09987*, 2025. 4

[4] Dailan He, Ziming Yang, Weikun Peng, Rui Ma, Hongwei Qin, and Yan Wang. Elic: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5718–5727, 2022. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[6] Zhaoyang Jia, Bin Li, Jiahao Li, Wenxuan Xie, Linfeng Qi, Houqiang Li, and Yan Lu. Towards practical real-time neural video compression. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12543–12552, 2025. 1, 2

[7] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 170–185, 2018. 2

[8] 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

[9] Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. *Advances in neural information processing systems*, 33:11913–11924, 2020. 1

[10] Linfeng Qi, Zhaoyang Jia, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Generative latent coding for ultra-low bitrate image and video compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025. 1, 2

[11] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 2

[12] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 2

[13] Ren Yang, Radu Timofte, and Luc Van Gool. Perceptual learned video compression with recurrent conditional gan. In *IJCAI*, pages 1537–1544, 2022. 1, 2