GEN2-VC: UNLOCKING GENERATIVE PRIORS FOR VIDEO COMPRESSION

Anonymous authors
Paper under double-blind review



Figure 1: (a) Gen2-VC better preserves source structure and details at the same or lower bitrates than distortion-oriented DCVC-UF (Li et al., 2026a), tokenizer-based GVC1D (Zheng et al., 2026), and diffusion-based YODA (Li et al., 2026b), as illustrated by HEVC-D examples. (b) Gen2-VC improves distortion–perception trade-offs across codecs through training-free transfer of adapters trained only on DCVC-UF, as illustrated on HEVC-E. BD-rates (%) ↓ are relative to VTM-23.0.

ABSTRACT

Under stringent bitrate constraints, existing video codecs struggle to balance source fidelity and perceptual realism. Distortion-oriented codecs often over-smooth details, while generative codecs risk introducing content and structural deviations and rely on codec-specific designs. This motivates a question: *Can existing codecs achieve a better distortion–perception trade-off through simple, reusable adaptation?* Our insight is that native codec reconstructions provide a shared interface through which generative refinement is anchored to source content while remaining decoupled from codec-specific representations. We therefore propose Gen2-VC, a generative video compression framework that enhances the outputs of learned and conventional codecs with a pretrained video prior, leaving their bitstreams and reference update processes unchanged. With the codec, VAE, and generative backbone frozen, lightweight LoRA adapters refine codec reconstruction through single-frame spatial adaptation followed by multi-frame temporal adaptation with the video prior. Using Wan2.1-T2V-1.3B, Gen2-VC-DCVC-UF outperforms previous leading codecs in LPIPS/DISTS and, to our knowledge, is the first generative video codec to surpass VTM-23.0 in both PSNR and MS-SSIM at low bitrates, based on BD-rates averaged over six datasets. Compared to VTM-23.0, it reduces bitrate by an average of 86.65% and 94.24% at matched LPIPS and DISTS, respectively. Adapters trained only on DCVC-UF improve perceptual quality on DCVC-RT, ECM, VTM, and HM without retraining.

1 INTRODUCTION

Learned video compression has achieved impressive coding performance through deep models that capture spatial structures and temporal dependencies in compact representations. However, stringent

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

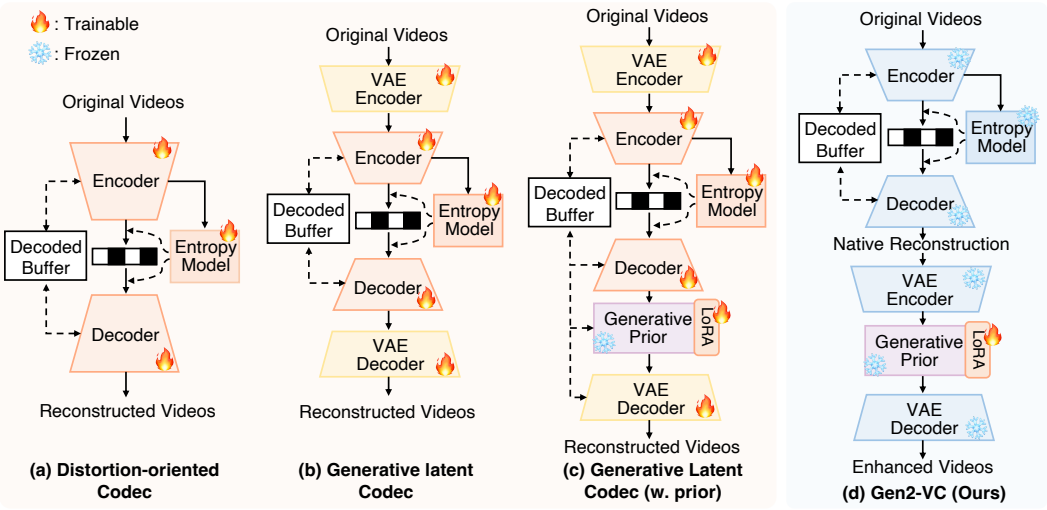


Figure 2: Comparison of video codec architectures. (a) Distortion-oriented coding reconstructs frames directly. (b) Generative latent coding uses perceptually or adversarially trained decoders. (c) Diffusion models refine decoded latents within conditional reconstruction pipelines. (d) Gen2-VC uses native reconstructions as a shared pixel-space interface, requiring no additional source-specific signals and preserving the codec's bitstream and reference updates.

bitrate constraints limit how much source information these representations can retain. Distortion-oriented codecs (Lu et al., 2019; Hu et al., 2021; Li et al., 2026a) prioritize pixel-level reconstruction accuracy, but can oversmooth textures and fine details, as illustrated in Figure 1(a). Even when coarse structures are preserved, missing details can leave reconstructed regions looking unnatural. This motivates codec optimization that accounts for perceptual realism alongside pixel-level fidelity.

Generative video codecs seek to mitigate oversmoothing and improve texture realism. One line compresses videos into generative latent representations and reconstructs them with perceptually or adversarially trained decoders (Yang et al., 2022; Qi et al., 2025; Guo et al., 2025; Zheng et al., 2026), while another incorporates diffusion models to synthesize plausible details (Mao et al., 2026; Xue et al., 2026; Li et al., 2026b; Xia et al., 2026). These perceptual gains can come at the cost of source fidelity, as illustrated by the content and structural deviations in Figure 1(a). Balancing fidelity and realism under stringent bitrate constraints therefore remains challenging (Blau & Michaeli, 2019). As illustrated by the representative architectures in Figure 2(b,c), existing designs also tie generation to codec-specific latent representations and conditional reconstruction pipelines, limiting reuse across codecs without redesign or retraining. This raises the question: *Can existing codecs achieve a better distortion–perception trade-off through simple, reusable adaptation?*

Video generation transports noise toward the natural video distribution, whereas compression maps source videos to a reconstruction distribution shaped by bitrate constraints. Our insight is to connect these processes by adapting a pretrained video prior to transport native codec reconstructions back toward the natural video distribution. For distortion-oriented codecs, these reconstructions retain basic source content and spatial layout, anchoring the transport to source videos so that refinement improves naturalness while maintaining source fidelity. These reconstructions provide a shared pixel-space interface, decoupling generative refinement from codec-specific representations and reference updates, which supports lightweight adaptation and reuse across codecs.

Building on this insight, we introduce **Gen2-VC**, which adapts a pretrained video prior while preserving the codec's native reference update chain, as illustrated in Figure 2(d). With the codec, VAE, and generative backbone frozen, we train lightweight LoRA adapters (Hu et al., 2022) in two stages using source video supervision: an intra-frame adapter (I-adapter) learns spatial refinement, then remains fixed while a predictive adapter (P-adapter) learns temporal refinement. At inference, the I-adapter refines frames independently, and the P-adapter jointly refines overlapping windows using its previous outputs as context. Each adapter call performs one near-terminal latent update.

Using Wan2.1-T2V-1.3B (Wan Team et al., 2025), Gen2-VC-DCVC-UF achieves a better distortion–perception trade-off at low bitrates. To our knowledge, it is the first generative video codec to achieve leading LPIPS (Zhang et al., 2018) and DISTS (Ding et al., 2022) performance while surpassing VTM-23.0 (VTM-23.0, 2025) in PSNR and MS-SSIM (Wang et al., 2003), measured using BD-rates averaged over six datasets in this regime. Training-free transfer further delivers perceptual gains on DCVC-RT (Jia et al., 2025), ECM (ECM-19.0, 2026), VTM (VTM-23.0, 2025), and HM (HM-18.0, 2023). Given the same codec inputs, Gen2-VC also achieves lower average BD-rates than the evaluated video restoration methods.

Our contributions are summarized as follows:

- We introduce a decoupled generative video compression framework that connects a shared video prior to learned and conventional codecs through native pixel-space reconstructions, preserving their bitstreams and reference update chains.
- We propose lightweight LoRA adaptation in spatial and temporal training stages, with one latent update per adapter call and the codec, VAE, and generative backbone frozen.
- Using one shared 1.3B video prior, we demonstrate improved rate–distortion–perception trade-offs at low bitrates across six datasets and training-free transfer of adapters trained only on DCVC-UF to other learned and conventional codecs.

2 Related Work

Distortion-oriented video compression. Conventional codecs (HM-18.0, 2023; VTM-23.0, 2025; ECM-19.0, 2026) combine handcrafted prediction, transform coding, and entropy coding under rate–distortion optimization. Among learned codecs, DVC (Lu et al., 2019) learns motion and residual coding, FVC (Hu et al., 2021) operates in feature space, and DCVC (Li et al., 2021) uses temporal context for conditional coding. DCVC-RT (Jia et al., 2025) and DCVC-UF (Li et al., 2026a) further improve coding efficiency and throughput. Gen2-VC uses their native reconstructions as a shared pixel-space interface for generative refinement, preserving existing coding pipelines.

Generative video compression. Generative codecs improve perceptual realism through two broad strategies. The first uses perceptual and adversarial training. PLVC (Yang et al., 2022) employs a recurrent conditional GAN, while GLC-Video (Qi et al., 2025) and GLVC (Guo et al., 2025) compress latents from a generative VQ-VAE and a pretrained video tokenizer, respectively. GVC1D (Zheng et al., 2026) exploits compact one-dimensional tokens with short- and long-term context, while GVC-RT (Dang et al., 2026) learns tokenizer-aligned continuous latents for real-time coding. The second uses pretrained diffusion and flow models. GNVC-VD (Mao et al., 2026) conditions VideoDiT on intermediate codec features to refine decoded latents. S^2VC (Xue et al., 2026) conditions diffusion on semantic context. YODA (Li et al., 2026b) applies temporal conditioning within its autoencoder and latent codec, while VoRTeC (Xia et al., 2026) combines latent coding with one-step flow refinement. These designs tie generation to codec-specific representations and reconstruction pipelines. Gen2-VC instead uses native pixel-space reconstructions as a shared interface after independent codec decoding, as illustrated in Figure 2(d).

Generative video restoration. Generative models support video detail recovery and super-resolution (He et al., 2024; Zhou et al., 2024; Xu et al., 2025). SeedVR (Wang et al., 2025) targets unknown real-world degradations, while SeedVR2 (Wang et al., 2026) enables efficient one-step restoration at high resolutions. Generic restoration training may not match codec-induced distortions or the fidelity–perception balance required at low bitrates. DiQP (Dehaghi et al., 2025) targets quantization distortions introduced by conventional video compression. Gen2-VC focuses on lightweight adaptation of a shared frozen video prior for low-bitrate rate–distortion–perception trade-offs, with training-free transfer across learned and conventional codecs.

3 Methods

Overview. An existing codec encodes a source video $X = \{x_t\}_{t=1}^{T}$ into a bitstream, then decodes it to obtain the native reconstruction $\widetilde{X} = \{\tilde{x}_t\}_{t=1}^{T}$. As illustrated in Figure 3(a), Gen2-VC first

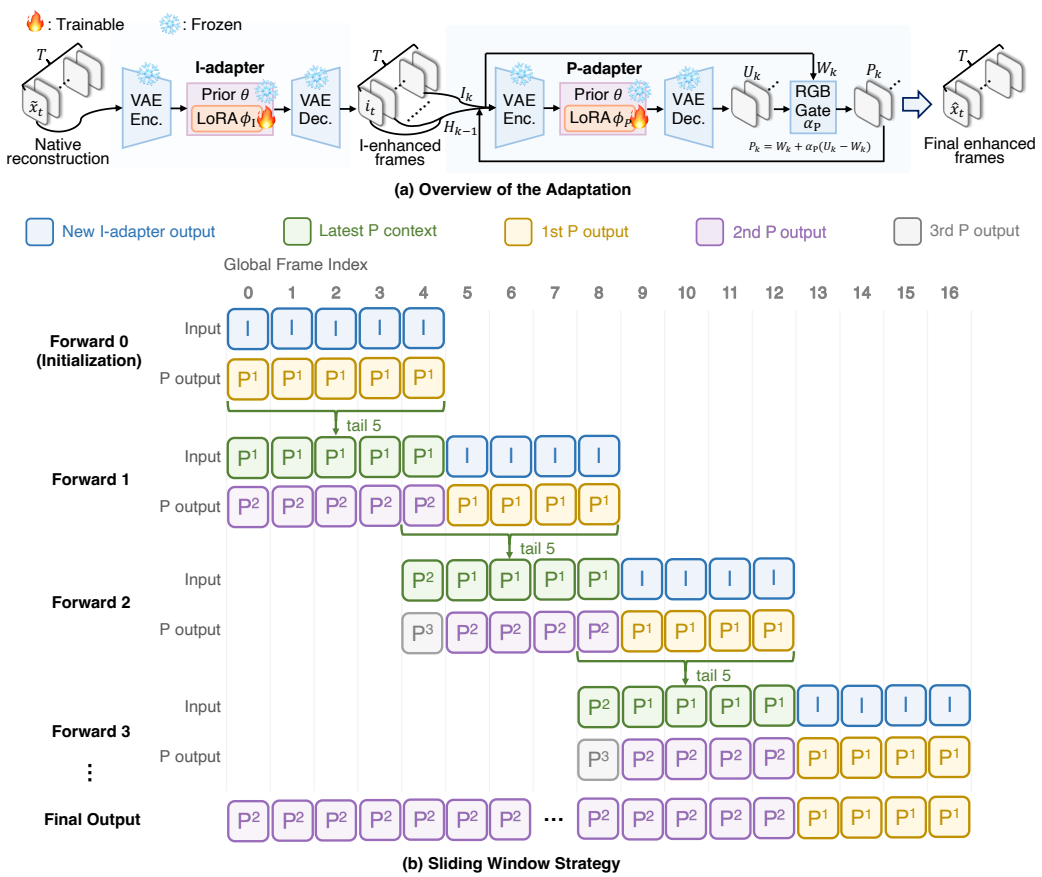


Figure 3: I-adapter and P-adapter in Gen2-VC. (a) The I-adapter restores individual frames before the P-adapter refines them jointly. (b) Sliding-window recurrence and output selection, shown with zero-based frame indices.

restores each frame of $\widetilde{X}$ independently with an intra-frame adapter (I-adapter), then jointly refines successive frames with a predictive adapter (P-adapter) to produce the final enhanced video $\hat{X} = \{\hat{x}_t\}_{t=1}^{T}$. The P-adapter reuses its outputs as recurrent context within Gen2-VC, leaving the codec's bitstream and reference update process unchanged. Both adapters share a frozen VAE and video prior and use separate LoRA parameters.

3.1 Source-Anchored Spatial Adaptation

Although entropy-constrained compression can remove texture and distort local structure, native reconstructions retain the source's core content and spatial layout, providing an anchor for detail recovery. The I-adapter independently refines each native reconstruction $\tilde{x}_t$ into an I-enhanced frame i_t, regardless of its codec frame type. Specifically, the frozen VAE encoder yields latent $z_t = \mathcal{E}(\tilde{x}_t)$. The video prior performs one update at a fixed near-terminal timestep $\tau_\star$ without added noise:

$$z_t^* = z_t - \beta_I \sigma_\star v_{\theta,\phi_I}(z_t; \tau_\star). \tag{1}$$

The frozen VAE decoder then produces the I-enhanced frame $i_t = \mathcal{D}(z_t^*)$. Here θ denotes the frozen backbone parameters, and ϕ_I denotes the LoRA (Hu et al., 2022) parameters. The predictor $v_{\theta,\phi_I}(\cdot; \tau_\star)$ provides the latent update signal under fixed conditioning shared across videos, requiring no source-specific auxiliary information. The coefficient $\sigma_\star$ scales this prediction at $\tau_\star$, and β_I is a fixed latent refinement strength for each operating point.

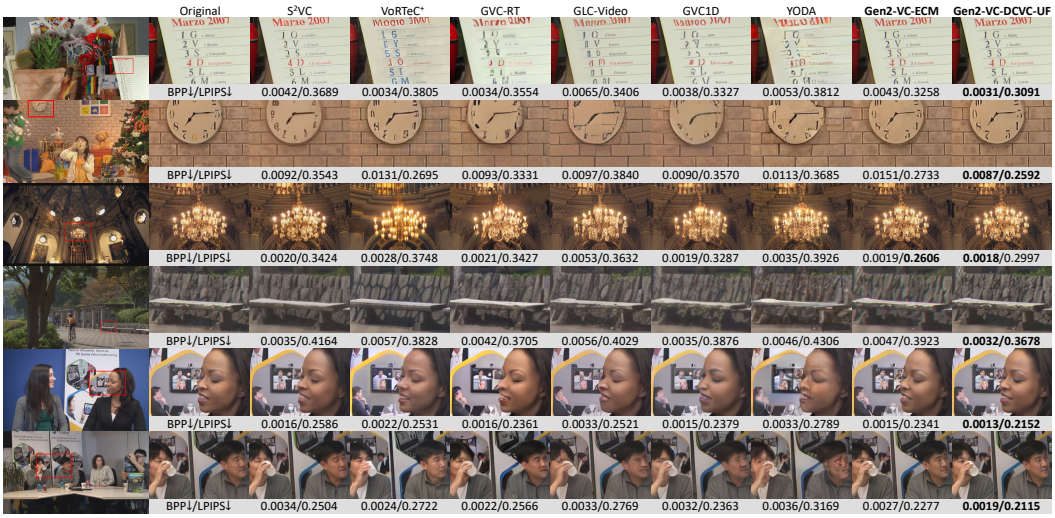


Figure 4: Gen2-VC better preserves source structure and details at low bitrates. Red boxes mark enlarged regions. BPP and LPIPS are reported below each reconstruction, with lower values preferred.

3.2 Recurrent Temporal Adaptation

Independent frame refinement does not explicitly exploit temporal correspondence. As illustrated in Figure 3(b), the P-adapter therefore jointly refines overlapping windows using its previous outputs as context. Specifically, recurrent call $k \geq 1$ receives $I_k = \{i_t\}_{t=t_k}^{t_k+L_{\text{in}}-1}$, containing L_{in} new I-enhanced frames starting at source index t_k. It combines these frames with history H_{k-1} from the preceding gated window output P_{k-1}:

$$H_{k-1} = \text{tail}_{L_{\text{tail}}}(P_{k-1}), \qquad W_k = \{H_{k-1}, I_k\}. \tag{2}$$

Here $\text{tail}_{L_{\text{tail}}}(\cdot)$ extracts the last L_{tail} frames, and braces denote temporal concatenation with history first, giving $L_{\text{win}} = L_{\text{tail}} + L_{\text{in}}$ frames. An initial call establishes history from I-enhanced frames.

The frozen VAE encoder yields latent $z_k^P = \mathcal{E}(W_k)$. With LoRA parameters ϕ_P and fixed strength β_P, the video prior performs one update at $\tau_\star$. The frozen VAE decoder produces a candidate window U_k, which the RGB gate $\alpha_{P,k}$ blends with W_k to obtain the complete output P_k:

$$\begin{aligned} U_k &= \mathcal{D}\big(z_k^P - \beta_P \sigma_\star v_{\theta,\phi_P}(z_k^P; \tau_\star)\big), \\ P_k &= W_k + \alpha_{P,k} \odot (U_k - W_k). \end{aligned} \tag{3}$$

Here $\odot$ denotes elementwise multiplication, and $\alpha_{P,k}$ controls refinement strength with first-visit and revisit scalars in $(0, 1]$, shared across pixels and windows. Both U_k and P_k retain all L_{win} frames of W_k. The gated window's tail forms the next history H_k, while overlapping calls provide successive estimates for each frame. Denoting frame t's j-th P-adapter estimate by $p_t^{(j)}$, we select $\hat{x}_t = p_t^{(2)}$ when available and $p_t^{(1)}$ otherwise, as illustrated in Figure 3(b). This finalizes $L_{\text{out}} = L_{\text{in}}$ new enhanced frames per call in steady state. More details are provided in Appendix A.1.

3.3 Staged Optimization

We train the adapters in two stages. First, we train the I-adapter to learn spatial refinement. We then freeze the trained I-adapter and train the P-adapter to learn temporal refinement. Throughout both stages, the codec, VAE, and pretrained backbone remain frozen. For an enhanced training frame y_t, either i_t or $p_t^{(j)}$, and the corresponding source frame x_t, the shared training objective is

$$\mathcal{L}(y_t, x_t) = \lambda_1 \|y_t - x_t\|_1 + \lambda_{\text{per}} \,\text{LPIPS}(y_t, x_t) + \lambda_{\text{adv}} \mathcal{L}_{\text{adv}}. \tag{4}$$

The ℓ_1 term constrains source fidelity, LPIPS (Zhang et al., 2018) measures perceptual differences, and the adversarial loss encourages natural appearance. For the I-adapter, we apply Eq. (4) independently to each frame with $\lambda_{\text{adv}} = 0.1$. For the P-adapter, we average this loss over all frames in each window with $\lambda_{\text{adv}} = 0.0$ and detach history between calls.

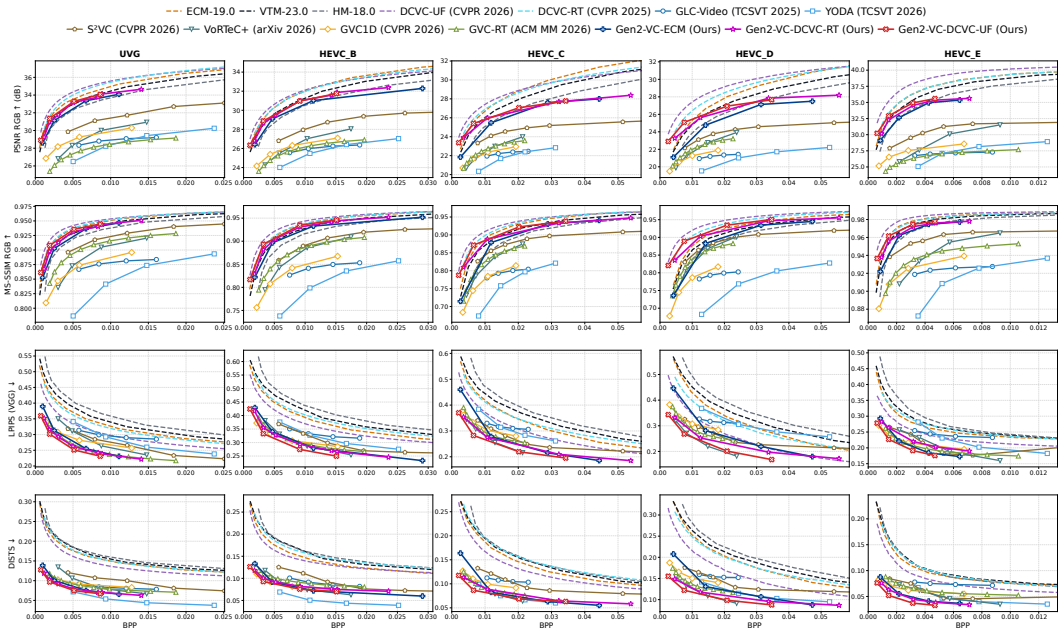


Figure 5: Rate–distortion–perception comparison on UVG and HEVC classes B/C/D/E. Higher is better for PSNR and MS-SSIM, while lower is better for LPIPS and DISTS. Bitrate is reported in bits per source pixel (BPP).

4 Experiments

4.1 Experimental Protocol

Training Details. We use Wan2.1-T2V-1.3B (Wan Team et al., 2025) as the frozen video prior and apply LoRA only to its self-attention Q/K/V/O projections. Both adapters use LoRA rank and scaling $(r, \alpha_{\mathrm{LoRA}}) = (16, 32)$ by default, each adding 0.42% of the backbone parameters per operating point. We first train the I-adapter on ImageNet (Deng et al., 2009) and OpenImages (Kuznetsova et al., 2020) at variable resolutions of 512×512–1024×1024, then the P-adapter on Vimeo (Xue et al., 2019) and OpenVid-HD (Nan et al., 2025) at 256×256–720×720. Training uses only DCVC-UF (Li et al., 2026a) reconstructions. For training-free transfer, we select among the four I/P-adapter pairs by the codec's quantization parameter, keeping weights and refinement settings fixed. Both stages use AdamW (Loshchilov & Hutter, 2019) on two NVIDIA RTX PRO 6000 GPUs, with a batch size of 16 and a learning rate decaying from 10^{-4} to 10^{-6}.

Testing Details. Following the DCVC series (Li et al., 2021; Jia et al., 2025; Li et al., 2026a), we evaluate the first 96 frames of each sequence in UVG (Mercat et al., 2020), MCL-JCV (Wang et al., 2016), and HEVC classes B/C/D/E (David et al., 2013), with intra-period -1. We convert YUV inputs to RGB using BT.709, following JPEG AI (Alshina et al., 2022), and compute PSNR, MS-SSIM (Wang et al., 2003), VGG-based LPIPS (Zhang et al., 2018), and DISTS (Ding et al., 2022) in RGB space. BD-rate (Bjontegaard, 2001) uses PCHIP interpolation over the common quality range without extrapolation, where positive values indicate bitrate increases and negative values indicate bitrate savings at matched quality. Appendix A.2.2 reports patch-based FID (Heusel et al., 2017) and KID (Bińkowski et al., 2018) following GLC-Video (Qi et al., 2025), together with MUSIQ (Ke et al., 2021), NIQE (Mittal et al., 2013), and CLIP-IQA (Wang et al., 2023).

Baselines. Distortion-oriented baselines comprise HM-18.0 (HM-18.0, 2023), VTM-23.0 (VTM-23.0, 2025), and ECM-19.0 (ECM-19.0, 2026) under low-delay configurations, together with the learned codecs DCVC-RT (Jia et al., 2025) and DCVC-UF-HTL (Li et al., 2026a) (DCVC-UF hereafter). Generative baselines include the VAE/tokenizer-based codecs GLC-Video (Qi et al., 2025), GVC1D (Zheng et al., 2026), and GVC-RT (Dang et al., 2026), and the diffusion-based codecs

Table 1: BD-rate (%) ↓ relative to VTM-23.0, averaged equally over UVG, MCL-JCV, HEVC-B, HEVC-C, HEVC-D, and HEVC-E. Missing coverage is shown as "–".

Method	PSNR	MS-SSIM	LPIPS	DISTS
Distortion-oriented video codec				
HM-18.0	38.32	43.93	29.25	11.55
VTM-23.0 (Anchor)	0.00	0.00	0.00	0.00
ECM-19.0	−20.95	−19.05	−19.77	−15.28
DCVC-RT	−21.41	−22.69	−18.42	−1.11
DCVC-UF	**−37.16**	**−40.04**	−49.41	−36.39
VAE/tokenizer-based video codec				
GLC-Video	–	365.05	−56.27	−78.87
GVC1D	403.37	199.97	−81.91	−91.14
GVC-RT	816.32	123.22	−81.99	−89.38
Diffusion-based video codec				
YODA	984.31	703.65	−57.14	−90.64
S^2VC	307.75	126.36	−65.30	−77.58
VoRTeC+	409.33	181.60	−80.13	−90.79
Video restoration methods				
SeedVR2 + DCVC-RT	–	360.94	−16.13	−33.16
VEnhancer-v2 + DCVC-RT	–	–	4.91	−28.04
DiQP + DCVC-RT	55.37	16.33	37.95	57.19
SeedVR2 + DCVC-UF	–	164.70	−44.51	−63.32
VEnhancer-v2 + DCVC-UF	–	–	−30.09	−48.18
DiQP + DCVC-UF	5.60	−18.74	−9.16	8.71
Gen2-VC variants				
Gen2-VC-HM	94.00	78.93	−21.26	−48.06
Gen2-VC-VTM	50.77	29.59	−66.33	−82.43
Gen2-VC-ECM	28.63	7.08	−78.33	−90.84
Gen2-VC-DCVC-RT	8.72	−7.28	−79.90	−91.76
Gen2-VC-DCVC-UF	−12.11	−28.03	**−86.65**	**−94.24**

YODA (Li et al., 2026b), S^2VC (Xue et al., 2026), and VoRTeC+ (Xia et al., 2026). We evaluate restoration models SeedVR2 (Wang et al., 2026) with wavelet color correction, DiQP (Dehaghi et al., 2025), and VEnhancer-v2 (He et al., 2024) on native DCVC-RT/DCVC-UF reconstructions.

4.2 Experimental Results

Quantitative Evaluation. Figure 5 shows rate–distortion–perception curves. Gen2-VC-DCVC-UF substantially outperforms the evaluated generative codecs in PSNR/MS-SSIM across all datasets. On six-dataset averages at low bitrates in Table 1, it saves 86.65%/94.24% bitrate at matched LPIPS/DISTS and 12.11%/28.03% at matched PSNR/MS-SSIM relative to VTM-23.0. To our knowledge, it is the first generative video codec to combine state-of-the-art LPIPS/DISTS performance with PSNR/MS-SSIM gains over VTM-23.0 on this six-dataset average. Training-free transfer of the same DCVC-UF-trained adapters to DCVC-RT, ECM, VTM, and HM substantially lowers mean LPIPS/DISTS BD-rates without modifying their bitstreams or reference update chains, supporting reusable adaptation across learned and conventional codecs.

Using the same codec inputs, Gen2-VC also outperforms SeedVR2, DiQP, and VEnhancer-v2 in the available full-reference comparisons in Table 1. Appendices A.2.2 and A.6 provide additional metrics and per-dataset BD-rates. SeedVR2 and VEnhancer-v2 attain favorable MUSIQ/CLIP-IQA scores despite weaker full-reference performance, suggesting a mismatch between restoration objectives and low-bitrate fidelity–perception requirements.

Qualitative Evaluation. Figures 1(a) and 4 illustrate why fidelity and perceptual quality should be assessed jointly at low bitrates. DCVC-UF can oversmooth details despite strong PSNR/MS-SSIM, whereas GVC1D and YODA can distort source content despite perceptual gains. Gen2-VC-DCVC-UF preserves structure and details more faithfully at the same or lower bitrates. Our user study

Table 2: Computational complexity on one A100 80 GB GPU. Video restoration methods and Gen2-VC report enhancement-only costs, with I- and P-adapters profiled separately. FPS: frames per second. TMAC/frame: Tera multiply-accumulate operations/frame. Memory: peak allocated/reserved (A/R) GPU memory, in GiB. Params.: total (trainable) parameters of default configurations, in M. "–" denotes the encoding FPS decided by the selected codec.

		FPS ↑		TMAC/frame ↓		Memory ↓	Params. ↓
Method	Res.	Enc.	Dec.	Enc.	Dec.	A/R	Total (Train.)
Diffusion-based video codec							
YODA	240p	3.958	5.011	2.419	1.601	9.85/10.49	2055.13
	480p	2.860	3.920	4.469	2.951	11.64/12.93	
	720p	1.536	2.292	10.279	6.776	16.69/21.37	
	1080p	0.780	1.230	21.813	14.370	26.73/39.32	
S^2VC	240p	4.857	5.600	0.918	0.859	7.32/8.09	1842.29
	480p	2.148	2.483	3.532	3.310	8.25/9.41	
	720p	0.987	1.113	8.650	8.138	9.94/12.50	
	1080p	0.404	0.439	20.490	19.403	13.31/18.61	
VoRTeC+	240p	9.341	6.312	0.344	0.893	9.24/10.01	1802.72
	480p	0.977	0.482	1.288	3.551	14.62/17.54	
	720p	0.476	0.240	3.011	9.271	24.18/29.76	
	1080p	0.188	0.097	6.561	24.290	43.20/56.00	
Video restoration methods							
SeedVR2	240p	–	0.368	–	2.014	36.36/36.74	3642.12
	480p	–	0.344	–	8.028	39.94/40.55	
	720p	–	0.258	–	18.586	46.87/49.11	
	1080p	–	0.194	–	42.515	68.37/78.09	
DiQP	240p	–	3.050	–	0.589	1.48/2.15	79.37
	480p	–	1.695	–	1.178	1.48/2.15	
	720p	–	0.527	–	3.534	1.48/2.15	
	1080p	–	0.261	–	7.067	1.48/2.15	
VEnhancer-v2	240p	–	0.072	–	145.383	19.55/78.62	2496.59
	480p	–	0.072	–	145.383	19.58/78.62	
	720p	–	0.071	–	145.383	19.63/78.62	
	1080p	–	0.093	–	216.563	23.81/78.62	
Gen2-VC variants							
I-adapter	240p	–	16.659	–	2.071	3.27/5.42	1551.79 (5.90)
	480p	–	5.234	–	8.205	3.29/12.56	
	720p	–	2.274	–	20.268	3.34/24.20	
	1080p	–	0.887	–	53.519	10.83/27.21	
P-adapter	240p	–	12.807	–	2.430	5.21/7.37	1551.79 (5.90)
	480p	–	3.553	–	10.396	11.22/19.35	
	720p	–	1.477	–	27.122	21.67/39.08	
	1080p	–	0.557	–	77.546	45.25/69.17	

in Appendix A.4 further supports Gen2-VC's gains in subjective quality. Appendix A.7 provides further codec and restoration comparisons.

Complexity Analysis. Table 2 profiles I/P separately, each adding 5.90M trainable parameters to the shared frozen VAE and backbone. At every tested resolution, both stages have higher FPS than S^2VC, VoRTeC+, and all three restoration methods, fewer MACs than VEnhancer-v2, and less peak GPU memory than SeedVR2. These component-level comparisons follow the profiling scopes detailed in Appendix A.5, which also reports the complete DCVC-UF-based decoding pipeline. The full pipeline outpaces VoRTeC+ and the restoration methods on one GPU and supports multi-GPU pipelining by keeping refinement outside codec reference updates. Gen2-VC thus combines parameter-efficient adaptation with competitive enhancement throughput while preserving native encoding efficiency, with all additional computation confined to decoder-side refinement.

4.3 Ablation Study

We select settings by jointly considering LPIPS/DISTS and LoRA parameter cost, while retaining PSNR/MS-SSIM gains over VTM-23.0, measured by BD-rates averaged over six datasets.

Table 3: Stage and rank ablations on HEVC-E using DCVC-UF and four adapter operating points. BD-rate (%) ↓ relative to VTM-23.0.

Variant	P output / rank	Total LoRA (M)	PSNR	MS-SSIM	LPIPS	DISTS
(a) Stages and output selection						
Native DCVC-UF	–	–	**-52.88**	**-46.08**	-53.05	-42.83
I-only	–	5.90	-7.99	-18.06	-88.35	-96.52
I+P	*First*	11.80	-8.95	-18.70	-88.73	-96.75
I+P	*Second*	11.80	-12.89	-22.51	**-92.06**	**-97.47**
pixel-P (no I)	*First*	11.80	-43.01	-38.00	-78.36	-87.65
pixel-P (no I)	*Second*	11.80	-37.09	-34.09	-81.04	-89.55
(b) I-adapter rank						
I-only	$r_I = 8$	2.95	-10.61	**-22.51**	-84.48	**-96.62**
I-only (default)	$r_I = 16$	5.90	-7.99	-18.06	-88.35	-96.52
I-only	$r_I = 32$	11.80	**-10.89**	-21.01	**-92.98**	-96.59
(c) P-adapter rank: fixed $r_I = 16$, Second outputs						
I+P (default)	$r_P = 16$	11.80	**-12.89**	**-22.51**	-92.06	-97.47
I+P	$r_P = 32$	17.69	-12.72	-21.47	**-92.20**	-97.48
I+P	$r_P = 64$	29.49	-12.72	-21.46	-92.18	**-97.51**

Effects of the P-Stage and Output Selection. We test recurrent refinement by comparing *I-only*, which applies the I-adapter framewise, with *I+P*, which adds the P-adapter. Across overlapping windows, *First* retains each frame's first P estimate, while *Second* uses its second when available and otherwise the first, following Figure 3(b). In Table 3(a), *Second* improves all four BD-rates over *I-only* and *First*, lowering LPIPS/DISTS from *I-only*'s $-88.35\%/-96.52\%$ to $-92.06\%/-97.47\%$. FloLPIPS (Danier et al., 2022) results in Appendix A.3 and cross-frame examples in Appendix A.7.2 further support these temporal refinement gains over the evaluated 96-frame clips.

Effectiveness of Staged Adaptation. To test adapter separation, *pixel-P* uses one rank-32 adapter trained first on single frames and then on multiple frames to refine native reconstructions directly. Its single- and multi-frame stages share the hardware and dataset mixtures used to train the I- and P-adapters, respectively, and both pipelines use 11.80M LoRA parameters. With *Second* outputs, *I+P* improves LPIPS/DISTS BD-rates from *pixel-P*'s $-81.04\%/-89.55\%$ to $-92.06\%/-97.47\%$ in Table 3(a). Under fixed GPU memory, multi-frame training uses lower resolutions than single-frame training, as described in Section 4.1. Appendix A.3 shows that *pixel-P* can lose perceptual quality at higher test resolutions despite spatial pretraining, consistent with this train–test mismatch. *I+P* preserves higher-resolution spatial refinement through the fixed I-adapter's outputs, allowing the P-adapter to focus on temporal modeling and supporting parameter-efficient staged adaptation.

LoRA Rank and Adaptation Capacity. Table 3(b) compares I-adapter ranks 8, 16, and 32 under a common training recipe. Considering LPIPS and DISTS jointly, we select $r_I = 16$ over rank 8. Rank 32 offers further gains but doubles the LoRA parameters. Under the same GPU memory budget for training, $r_P = 16$ offers the best PSNR/MS-SSIM and comparable LPIPS/DISTS in Table 3(c), with 11.80M total LoRA parameters versus 17.69M/29.49M for ranks 32/64. Six-dataset averages in Appendix A.3.2 further support $r_P = 16$ as our default.

5 Conclusion

We presented Gen2-VC, a decoupled framework that connects a pretrained video prior to existing codecs through native pixel-space reconstructions, anchoring refinement to source content while preserving bitstreams and native reference update chains. Lightweight LoRA adapters learn spatial and temporal refinement in stages, with one latent update per adapter call and the codec, VAE, and backbone frozen. Experiments with one shared Wan2.1-T2V-1.3B prior demonstrate improved rate–distortion–perception trade-offs at low bitrates across six datasets. Adapters trained only on DCVC-UF further deliver perceptual gains on other learned and conventional codecs through training-free transfer. Gen2-VC thus provides a practical route to incorporate generative refinement into existing video compression systems without redesigning their coding pipelines.

AI USE STATEMENT

The authors confirm that no AI-assisted tools were used in the preparation of this manuscript.

REPRODUCIBILITY STATEMENT

Sections 3 and 4 describe the method and experimental protocol. Appendices A.1, A.2, and A.5 provide implementation details, evaluation settings, and profiling procedures. We will publicly release the code and model weights upon acceptance.

REFERENCES

E Alshina, J Ascenso, T Ebrahimi, F Pereira, and T Richter. AHG 11. *Brief information about JPEG AI CfP status. In JVET-AA0047*, 6:13, 2022.

Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations (ICLR)*, 2018.

Gisle Bjontegaard. Calculation of average psnr differences between rd-curves. *ITU SG16 Doc. VCEG-M33*, 2001.

Yochai Blau and Tomer Michaeli. Rethinking lossy compression: The rate-distortion-perception tradeoff. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97, pp. 675–685. PMLR, 2019.

Tianjian Dang, Sixian Wang, Lei Luo, Guo Lu, and Jincheng Dai. GVC-RT: Towards real-time generative video compression at ultra-low bitrates. *arXiv preprint arXiv:2608.04891*, 2026.

Duolikun Danier, Fan Zhang, and David Bull. FloLPIPS: A bespoke video quality metric for frame interpolation. In *2022 Picture Coding Symposium (PCS)*, pp. 283–287, 2022.

F David, S Karl, and R Chris. Doc. JCTVC-N1006: Common test conditions and software reference configurations for HEVC range extensions. *Joint Collaborative Team on Video Coding (JCT-VC) ITU-T SG*, 2013.

Ali Mollaahmadi Dehaghi, Reza Razavi, and Mohammad Moshirpour. Reversing the damage: A QP-aware transformer-diffusion approach for 8k video restoration under codec compression. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1258–1267, 2025.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, 2022.

ECM-19.0. Enhanced compression model (ECM) reference software, 2026. URL `https://vcgit.hhi.fraunhofer.de/ecm/ECM/-/tags/ECM-19.0`. Accessed: 2026-09-08.

Zongyu Guo, Zhaoyang Jia, Jiahao Li, Xiaoyi Zhang, Bin Li, and Yan Lu. Generative latent video compression. *arXiv preprint arXiv:2510.09987*, 2025. URL `https://arxiv.org/abs/2510.09987`.

Jingwen He, Tianfan Xue, Dongyang Liu, Xinqi Lin, Peng Gao, Dahua Lin, Yu Qiao, Wanli Ouyang, and Ziwei Liu. VEnhancer: Generative space-time enhancement for video generation. *arXiv preprint arXiv:2407.07667*, 2024.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

HM-18.0. HEVC test model (HM) reference software, 2023. URL https://vcgit.hhi.fraunhofer.de/jvet/HM/-/tags/HM-18.0. Accessed: 2026-09-08.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

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)*, pp. 1502–1511, 2021.

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 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12543–12552, 2025.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5148–5157, 2021.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, March 2020.

Jiahao Li, Bin Li, and Yan Lu. Deep contextual video compression. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 18114–18125, 2021.

Jiahao Li, Wenxuan Xie, Zhaoyang Jia, Bin Li, Zongyu Guo, Xiaoyi Zhang, and Yan Lu. Ultra-fast neural video compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 41311–41321, 2026a.

Xingchen Li, Junzhe Zhang, Junqi Shi, Ming Lu, and Zhan Ma. YODA: Yet another one-step diffusion-based video compressor. *IEEE Transactions on Circuits and Systems for Video Technology*, 2026b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

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 (CVPR)*, pp. 11006–11015, 2019.

Qi Mao, Hao Cheng, Tinghan Yang, Libiao Jin, and Siwei Ma. Generative neural video compression via video diffusion prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 43239–43248, 2026.

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*, pp. 297–302, 2020.

Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013.

Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. Openvid-1M: A large-scale high-quality dataset for text-to-video generation. In *International Conference on Learning Representations (ICLR)*, pp. 1045–1064, 2025.

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*, 35(10):10500–10515, May 2025.

VTM-23.0. VVC test model (VTM) reference software, 2025. URL https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/tags/VTM-23.0. Accessed: 2026-09-08.

Wan Team, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. URL https://arxiv.org/abs/2503.20314.

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)*, pp. 1509–1513, 2016.

Jianyi Wang, Kelvin C. K. Chan, and Chen Change Loy. Exploring CLIP for assessing the look and feel of images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 2555–2563, 2023. doi: 10.1609/aaai.v37i2.25353. URL https://ojs.aaai.org/index.php/AAAI/article/view/25353.

Jianyi Wang, Zhijie Lin, Meng Wei, Yang Zhao, Ceyuan Yang, Chen Change Loy, and Lu Jiang. SeedVR: Seeding infinity in diffusion transformer towards generic video restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2161–2172, 2025.

Jianyi Wang, Shanchuan Lin, Zhijie Lin, Yuxi Ren, Meng Wei, Zongsheng Yue, Shangchen Zhou, Hao Chen, Yang Zhao, Ceyuan Yang, Xuefeng Xiao, Chen Change Loy, and Lu Jiang. SeedVR2: One-step video restoration via diffusion adversarial post-training. In *International Conference on Learning Representations (ICLR)*, 2026.

Z. Wang, E.P. Simoncelli, and A.C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, volume 2, pp. 1398–1402, 2003.

Yichong Xia, Qinhong Wu, Jinpeng Wang, Zeyuan Chen, and Haoqian Wang. VoRTeC: Taming foundation flow for one-step real time video compression. *arXiv preprint arXiv:2609.02291*, 2026. URL https://arxiv.org/abs/2609.02291.

Yiran Xu, Taesung Park, Richard Zhang, Yang Zhou, Eli Shechtman, Feng Liu, Jia-Bin Huang, and Difan Liu. VideoGigaGAN: Towards detail-rich video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2139–2149, 2025.

Naifu Xue, Zhaoyang Jia, Jiahao Li, Bin Li, Zihan Zheng, Yuan Zhang, and Yan Lu. Single-step diffusion-based video coding with semantic-temporal guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9752–9761, 2026.

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

Ren Yang, Radu Timofte, and Luc Van Gool. Perceptual learned video compression with recurrent conditional GAN. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1537–1544, 2022.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 2018.

Zihan Zheng, Zhaoyang Jia, Naifu Xue, Jiahao Li, Bin Li, Zongyu Guo, Xiaoyi Zhang, Zhenghao Chen, Houqiang Li, and Yan Lu. Generative video compression with one-dimensional latent representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 41256–41265, 2026.

Shangchen Zhou, Peiqing Yang, Jianyi Wang, Yihang Luo, and Chen Change Loy. Upscale-A-Video: Temporal-consistent diffusion model for real-world video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2535–2545, 2024.

A APPENDIX

The appendix is organized as follows.

Appendix A.1 Implementation, training, and recurrent inference.
Appendix A.2 Codec configurations, color conversion, and metric implementations.
Appendix A.2.2 Additional quality metrics.
Appendix A.2.3 Extended codec and restoration comparisons.
Appendix A.3 Stage, output-selection, and LoRA-rank ablations.
Appendix A.4 User-study interface, protocol, and preference results.
Appendix A.5 Profiling protocol, component costs, and full decoding pipelines.
Appendix A.6 Per-dataset BD-rates and aggregation rules.
Appendix A.7 Codec, stage, and restoration visual comparisons.

A.1 IMPLEMENTATION AND TRAINING DETAILS

Gen2-VC takes native RGB reconstructions as its pixel-space interface and uses the frozen VAE to encode and decode refinement latents.

Training Configuration. Table 4 summarizes training settings, adapter sizes, and refinement controls. The codec, VAE, and backbone remain frozen, as does the I-adapter during P-stage training. LoRA targets self-attention Q/K/V/O projections in all 30 backbone blocks, with no dropout. Rank comparisons appear in Appendix A.3.2.

Table 4: Training settings, default adapter sizes, and refinement controls.

Setting	I-adapter	P-adapter
Training data	ImageNet, OpenImages	Vimeo, OpenVid-HD
Spatial resolution	512×512–1024×1024	256×256–720×720
Training codec	DCVC-UF	
LoRA variables	ϕ_I	ϕ_P
LoRA $(r, \alpha_{\mathrm{LoRA}})$	$(16, 32)$	$(16, 32)$
LoRA parameters	5,898,240 (0.42%)	5,898,240 (0.42%)
$(\lambda_1, \lambda_{\mathrm{per}}, \lambda_{\mathrm{adv}})$	$(1.0, 0.8, 0.1)$	$(1.0, 0.8, 0.0)$
Discriminator	PatchGAN (training only)	None
Optimizer	AdamW	
Learning rate	10^{-4} decaying to 10^{-6}	
Batch size	16	
Training iterations	6,000	1,000
Hardware	Two NVIDIA RTX PRO 6000 GPUs	
Refinement strength	$(1.00, 0.85, 0.70, 0.60)$	$(1.00, 0.70, 0.35, 0.25)$
RGB gate (first/revisit)	–	0.02 / 0.09
$(L_{\mathrm{win}}, L_{\mathrm{tail}}, L_{\mathrm{in}}, L_{\mathrm{out}})$	–	$(9, 5, 4, 4)$
Initial context	–	Five I-enhanced frames
Training inputs	Native reconstructions	Fixed I-enhanced frames
Loss aggregation	Per frame	Mean over all window frames
History gradients	–	Detached between calls
Inference	Framewise	Section 3.2, Figure 3(b)

Each of the four DCVC-UF operating points has a separately trained I/P-adapter pair. LoRA counts are per operating point, with percentages relative to the 1,418,996,800 frozen backbone parameters, excluding the VAE and codec. Refinement strengths and RGB gates are set empirically before training to control refinement at different bitrates and remain fixed during training and inference. Strength tuples follow increasing bitrate, while the two RGB gates are shared across all four points. Both controls transfer unchanged, without further tuning for individual codecs, datasets, videos, or

frames. At inference, we select among the four adapter pairs using the target codec's quantization parameter, keeping the selected weights and all other inference settings fixed.

One-update Enhancement. To align conditioning with the video prior, we follow Wan2.1's official inference example for the CFG scale of 6 and the negative prompt. Both adapters share an empty positive prompt and fixed, precomputed UMT5 embeddings across videos and bitrate points, without loading or running the text encoder at inference. CFG combines positive/negative predictions as $v = v_- + 6(v_+ - v_-)$. Index 49 (zero-based) of a 50-point Diffusers 0.39.0 `UniPCMultistepScheduler` schedule with 1,000 training timesteps and flow shift 3 gives $\sigma_\star \approx 0.06040539$ and $\tau_\star = \lfloor 1000\sigma_\star \rfloor = 60$. Each call scales RGB inputs to $[-1, 1]$, applies VAE posterior-mode encoding and Wan's latent normalization, then performs one latent update without added noise as in Eqs. (1) and (3), with the next noise level set to zero. We invert the normalization, decode, then rescale and clip outputs to $[0, 1]$.

> **Negative prompt:** Bright tones, overexposed, static, blurred details, subtitles, style, works, paintings, images, static, overall gray, worst quality, low quality, JPEG compression residue, ugly, incomplete, extra fingers, poorly drawn hands, poorly drawn faces, deformed, disfigured, misshapen limbs, fused fingers, still picture, messy background, three legs, many people in the background, walking backwards

A.2 Testing Protocol and Metric Implementations

Conventional codec configurations. Table 5 lists the low-delay configurations. Paths are relative to each codec's installation directory.

Table 5: Conventional codec executables and low-delay configurations.

Codec	Executable	Configuration file
HM-18.0	`TAppEncoder`	`cfg/encoder_lowdelay_main10.cfg`
VTM-23.0	`EncoderApp`	`cfg/encoder_lowdelay_vtm.cfg`
ECM-19.0	`EncoderApp`	`cfg/encoder_lowdelay_ecm.cfg`

Test command and evaluation. We encode the first 96 frames of each planar 8-bit YUV 4:2:0 sequence at its source resolution, setting `frame_number=96` and `intra_period=-1` below. Select the configuration from Table 5 and use `TAppEncoder` instead of `EncoderApp` for HM. The launcher's default `frame_rate` is 24.

```
EncoderApp -c {encoder_configuration.cfg} \
  -f {frame_number} -q {qp} --IntraPeriod={intra_period} \
  --InputFile={src_yuv} \
  --SourceWidth={width} --SourceHeight={height} \
  --FrameRate={frame_rate} --Level=6.2 --InputBitDepth=8 \
  --DecodingRefreshType=2 -b {output.bin} -o {enc.yuv}
```

The `-b` option writes `output.bin`, and `-o` writes the native YUV reconstruction `enc.yuv`. Bitrate is BPP $= 8|$`output.bin`$|_{\text{bytes}}/(96HW)$ for frame size $H \times W$. Following Table 6, we convert `src_yuv` and `enc.yuv` to RGB8 for evaluation. The converted native frames also serve as Gen2-VC inputs. Framewise metrics in Table 8 are averaged over the 96 frames within each sequence, then equally across sequences in each dataset.

Table 6: Source and native-reconstruction conversion to 8-bit RGB.

Step	Input convention and implementation
Read and normalize	NumPy reads planar YUV 4:2:0. Source samples are 8-bit and divided by 255. Native reconstructions are read as 10-bit samples in little-endian 16-bit words and divided by 1023.
Upsample chroma	SciPy `ndimage.zoom`, with `order=0`, performs nearest-neighbor $2\times$ upsampling to luma resolution.
Convert color	A custom NumPy implementation applies full-range BT.709 with $(K_R, K_G, K_B) = (0.2126, 0.7152, 0.0722)$, following the DCVC-RT transform for both source and reconstruction.
Quantize and save	`np.rint(np.clip(rgb,0,1)*255)` followed by `uint8` conversion produces RGB8. Pillow reads and writes the RGB PNGs used for evaluation.

Baseline availability and rates. GLVC (Guo et al., 2025) and GNVC-VD (Mao et al., 2026) are excluded from quantitative comparisons because their code was unavailable at evaluation time. HM, VTM, ECM, DCVC-RT, DCVC-UF, YODA, and GVC-RT use actual bitstream rates, whereas GLC-Video and GVC1D use entropy-model estimates. S^{2}VC combines estimated I-frame latent rates, excluding hyperlatents, with measured P-frame rANS bytes. VoRTeC+ estimates latent/AIN rates, including tail padding but excluding auxiliary metadata. Gen2-VC and restoration methods retain their input codec's bitrate.

DiQP evaluation. We evaluate DiQP using its highest quantization setting, typically associated with the lowest-bitrate end of its original codec range.

SeedVR2 color correction. We apply the wavelet color correction supported by the official SeedVR2 implementation to its quantized outputs, using the corresponding native reconstructions as color references. At identical bitrates, Table 7 shows improved six-dataset mean BD-rates for MS-SSIM, LPIPS, and DISTS on both codecs. We therefore include color correction in both quality evaluation and timing. PSNR is omitted because its quality ranges do not overlap VTM-23.0. BD-rate follows the common quality-sorting convention before PCHIP integration, including for non-monotonic curves.

Table 7: SeedVR2 with and without wavelet color correction. BD-rate (%) ↓ relative to VTM-23.0, averaged equally over UVG, MCL-JCV, HEVC-B, HEVC-C, HEVC-D, and HEVC-E.

Method	Color fix	MS-SSIM	LPIPS	DISTS
SeedVR2 + DCVC-RT	Without	461.07	-12.50	-8.98
SeedVR2 + DCVC-RT	With	**360.94**	**-16.13**	**-33.16**
SeedVR2 + DCVC-UF	Without	234.79	-41.77	-61.46
SeedVR2 + DCVC-UF	With	**164.70**	**-44.51**	**-63.32**

A.2.1 Metric Implementations

The main quantitative comparisons use saved, quantized RGB frames. Table 8 records the libraries and explicit call settings in the evaluation code. FID/KID pool patches from all frames and sequences at each dataset/rate point, rather than averaging framewise scores.

FID/KID use 64×64 patches when the dataset's minimum frame dimension is below 256, and 256×256 otherwise. Patches are extracted with `torch.nn.functional.unfold` at stride equal to patch size, using GLC's two-offset rule (`split_patch_num=2`) when the shifted grid fits. NIQE retries numerical linear-algebra failures with the same PyIQA implementation on CPU. Its averages use finite frames and then equally weight sequences with a defined mean.

Table 8: Metric libraries and evaluation settings.

Metric	Implementation	Key settings
PSNR	NumPy (custom implementation)	Float64 RGB MSE on $[0, 1]$, $-10 \log_{10}$ MSE. Perfect frames use a 120 dB cap.
MS-SSIM	`pytorch-msssim`	RGB in $[0, 1]$, unit data range, per-frame scores.
LPIPS	`lpips`	VGG backbone, RGB rescaled to $[-1, 1]$.
FloLPIPS	Official FloLPIPS implementation	AlexNet backbone and FP32 PWC optical flow. Average 95 adjacent-frame pairs per 96-frame sequence, then equally average sequences within each dataset.
DISTS	`DISTS-pytorch`	RGB in $[0, 1]$, evaluation mode.
FID / KID	TorchMetrics with `torch-fidelity`	Default constructors, uint8 GLC-style patches. KID reports the mean and uses PyTorch seed 0 per dataset/rate point.
MUSIQ	PyIQA, `musiq`	Reconstructed RGB in $[0, 1]$, default KonIQ checkpoint.
NIQE	PyIQA, `niqe`	Reconstructed RGB in $[0, 1]$, default model. Undefined values are excluded with coverage recorded.
CLIP-IQA	PyIQA, `clipiqa`	Reconstructed RGB in $[0, 1]$, default CLIP RN50 weights.
BD-rate	`bjontegaard` 1.3.0	PCHIP over the common quality interval, without extrapolation. MS-SSIM uses raw values, without dB conversion.

A.2.2 Additional Quality Metrics

Figure 6 shows that Gen2-VC substantially reduces FID and KID relative to native codec reconstructions across all six datasets. Lower NIQE and generally higher MUSIQ/CLIP-IQA at low bitrates further support its perceptual gains beyond LPIPS and DISTS.

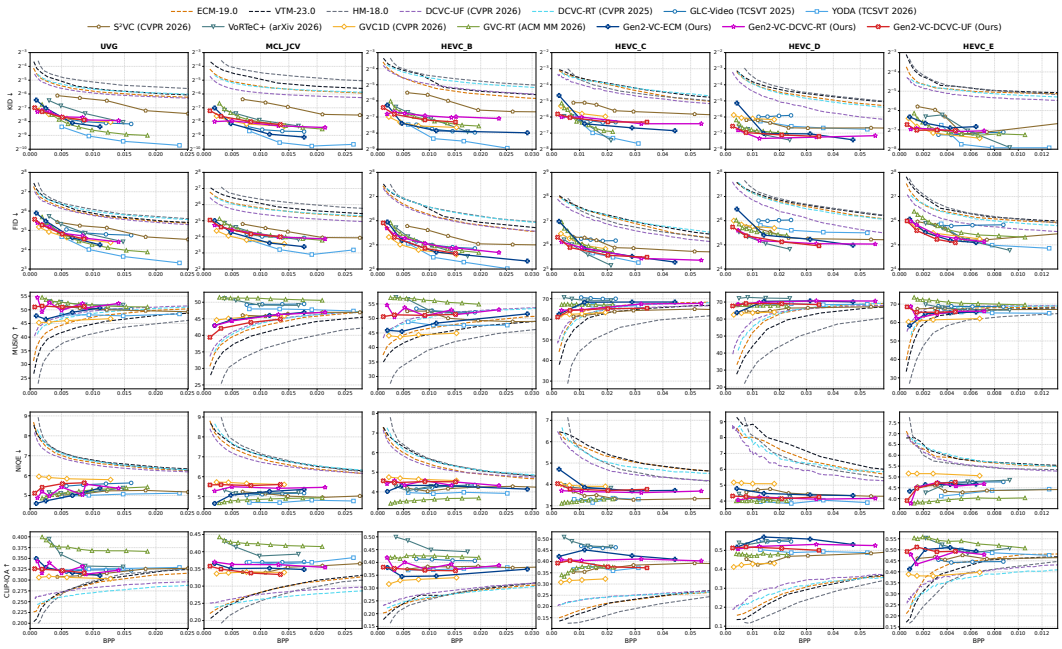


Figure 6: FID, KID, MUSIQ, NIQE, and CLIP-IQA comparisons with distortion-oriented and generative codecs.

A.2.3 Extended Codec and Restoration Comparisons

Figure 7 shows that adapters trained only on DCVC-UF improve LPIPS/DISTS on DCVC-RT, ECM, VTM, and HM without retraining. Given the same codec inputs, Gen2-VC also achieves a better fidelity–perception balance than the restoration baselines.

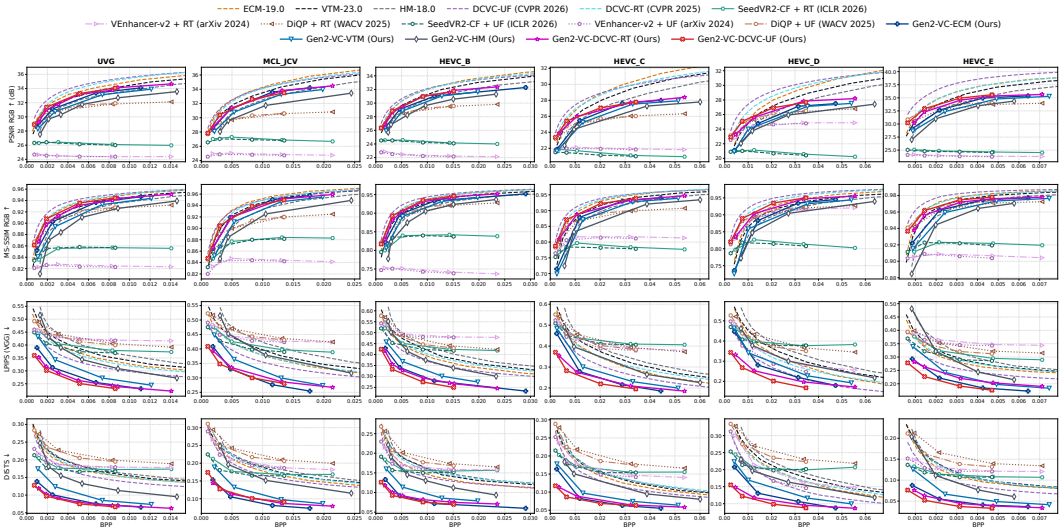


Figure 7: Full-reference comparisons across codec and restoration configurations.

Figure 8 shows substantially lower FID/KID for Gen2-VC than the restoration baselines given the same codec inputs. These gains complement its stronger full-reference performance, although SeedVR2 and VEnhancer-v2 attain higher MUSIQ or CLIP-IQA at some operating points.

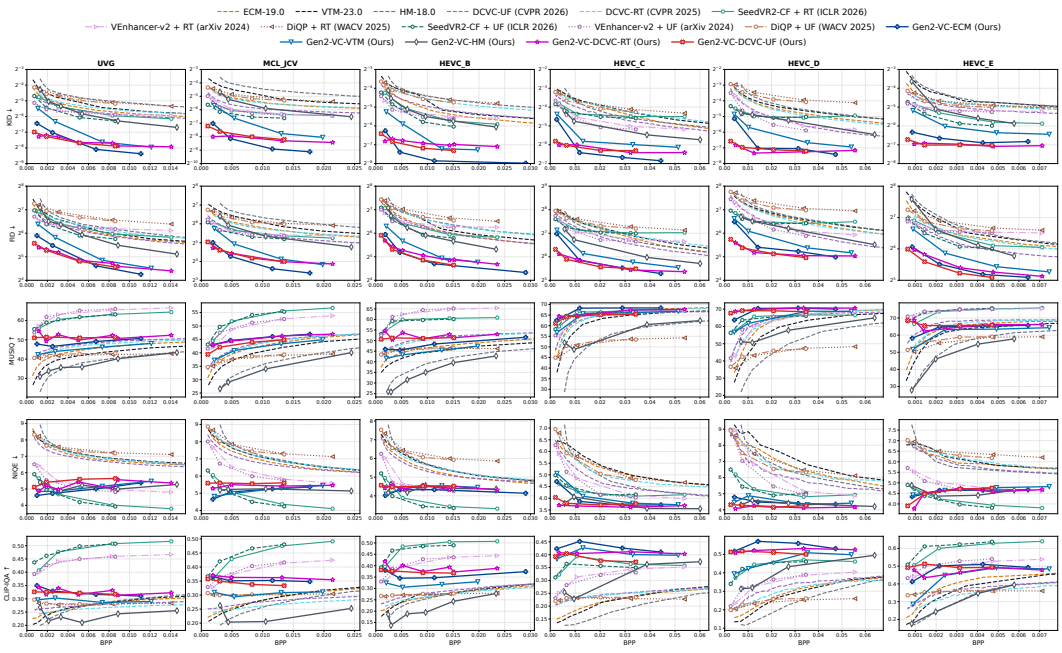


Figure 8: Training-free cross-codec transfer and comparisons with video restoration methods on FID, KID, MUSIQ, NIQE, and CLIP-IQA.

A.3 Additional Ablations

Table 9 complements the HEVC-E ablations in Table 3 with six-dataset averages. We select settings for the best joint LPIPS/DISTS performance while retaining negative six-dataset mean BD-rates in both PSNR and MS-SSIM relative to VTM-23.0 and accounting for LoRA parameter cost.

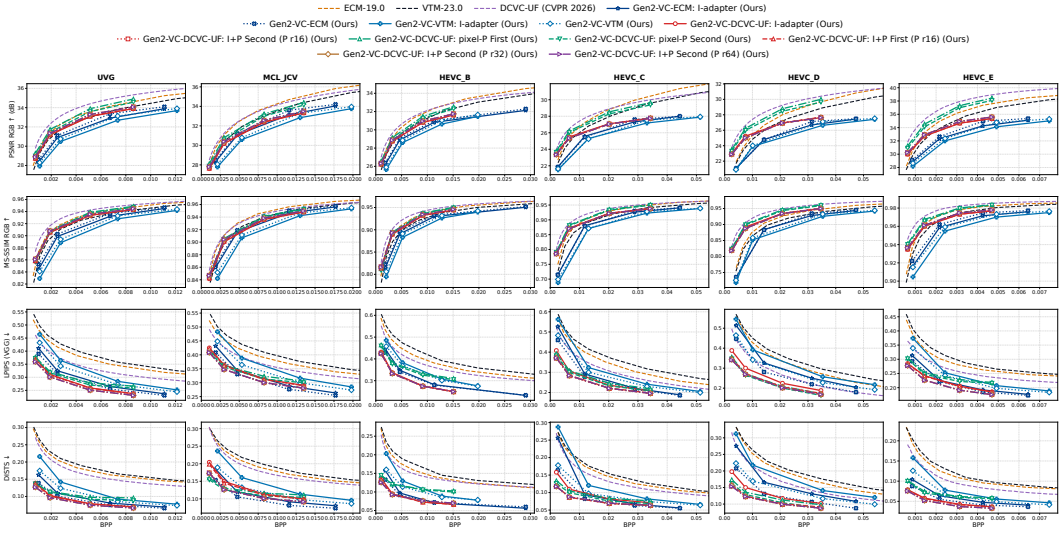


Figure 9: Stage and rank comparisons on six datasets. The default *I+P* and *pixel-P* have equal total LoRA parameters. At inference, *pixel-P* directly refines native reconstructions without a separate I-adapter.

A.3.1 STAGES AND OUTPUT SELECTION

Figure 9 compares *I-only*, which applies the I-adapter framewise, with *I+P*, which adds recurrent P-adapter refinement as in Figure 3(a). Following Figure 3(b), *First* retains each frame's first P estimate, whereas *Second* retains its second when available and the first otherwise. On HEVC-E, Table 3(a) shows that *I+P Second* improves all four BD-rates over *I-only* and *I+P First*. Table 9(a) confirms this trend in six-dataset averages. Table 10 further compares FloLPIPS BD-rates relative to native DCVC-UF on HEVC-D/C/B, UVG, and MCL-JCV. *I+P Second* achieves the lowest BD-rate on all five datasets, improving over both *I-only* and *First*. Together, these results support the P-stage's reconstruction and motion-sensitive perceptual gains. Cross-frame comparisons in Appendix A.7.2 provide complementary qualitative evidence of consistent detail recovery across sampled frames from the evaluated 96-frame clips, including after training-free transfer.

To assess adapter separation, *pixel-P* uses one rank-32 adapter for single-frame spatial training followed by multi-frame temporal training. At inference, it directly refines native reconstructions without a separate I-adapter. Its multi-frame stage shares the P-adapter's training hardware and dataset mixture, and both pipelines have 11.80M total LoRA parameters. With *Second* selection, *I+P* improves mean LPIPS/DISTS BD-rates from *pixel-P*'s $-81.97\%/-90.49\%$ to $-86.65\%/-94.24\%$. With *Second* outputs, *pixel-P* also has worse FloLPIPS BD-rates than *I+P* on all five datasets in Table 10, with the largest gap on HEVC-B. Under fixed GPU memory, multi-frame training uses lower resolutions than single-frame spatial training, as summarized in Table 4. Despite spatial pretraining, *pixel-P* can lose perceptual quality at higher test resolutions, consistent with this train–test resolution mismatch. *I+P* instead keeps the independently trained I-adapter fixed, preserving its higher-resolution spatial refinement in the I-enhanced inputs supplied to the P-adapter. This supports the parameter efficiency of separating spatial and temporal adaptation.

Table 9: Stage and rank ablations on DCVC-UF using four adapter operating points. BD-rate (%) ↓ relative to VTM-23.0, averaged equally over UVG, MCL-JCV, HEVC-B, HEVC-C, HEVC-D, and HEVC-E.

Variant	P output / rank	Total LoRA (M)	PSNR	MS-SSIM	LPIPS	DISTS
(a) Stages and output selection						
Native DCVC-UF	–	–	**-37.16**	**-40.04**	-49.41	-36.39
I-only	–	5.90	-9.19	-24.93	-83.38	-90.93
I+P	*First*	11.80	-8.72	-24.93	-85.03	-93.26
I+P	*Second*	11.80	-12.11	-28.03	**-86.65**	**-94.24**
pixel-P (no I)	*First*	11.80	-31.95	-34.61	-80.12	-89.53
pixel-P (no I)	*Second*	11.80	-25.26	-31.80	-81.97	-90.49
(b) I-adapter rank						
I-only	$r_I = 8$	2.95	-9.53	-25.22	-82.64	**-91.13**
I-only (default)	$r_I = 16$	5.90	-9.19	-24.93	-83.38	-90.93
I-only	$r_I = 32$	11.80	**-10.72**	**-26.18**	**-84.34**	-90.99
(c) P-adapter rank: fixed $r_I = 16$, Second outputs						
I+P (default)	$r_P = 16$	11.80	**-12.11**	**-28.03**	**-86.65**	**-94.24**
I+P	$r_P = 32$	17.69	-11.62	-27.25	-86.61	-94.19
I+P	$r_P = 64$	29.49	-11.59	-27.20	-86.60	-94.18

Table 10: FloLPIPS BD-rate (%) ↓ relative to native DCVC-UF, using four adapter operating points and 96-frame clips. HEVC-E is omitted because the *I-only* and *I+P* curves have no overlapping quality range with the anchor.

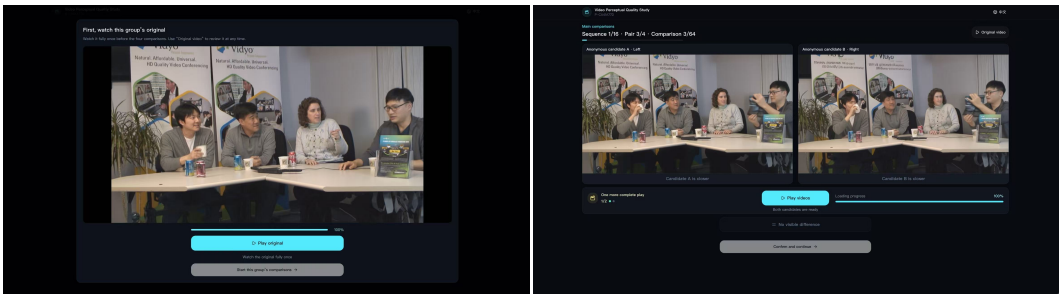
Variant	P output	HEVC-D	HEVC-C	HEVC-B	UVG	MCL-JCV
Native DCVC-UF (Anchor)	–	0.00	0.00	0.00	0.00	0.00
I-only	–	-54.60	-54.71	-76.36	-83.76	-63.65
I+P	*First*	-66.28	-73.83	-77.87	-84.56	-67.12
I+P	*Second*	**-68.02**	**-76.15**	**-81.71**	**-85.48**	**-77.37**
pixel-P (no I)	*Second*	-67.09	-72.31	-67.38	-82.24	-76.70

A.3.2 LoRA Rank

Table 9(b) supports $r_I = 16$ as a compromise between joint LPIPS/DISTS performance and parameter count. Rank 32 improves both perceptual metrics over rank 16 but doubles its LoRA parameters. Figure 9 and Table 9(c) compare P-adapter configurations at ranks 16, 32, and 64, with $r_I = 16$, $\alpha_{\mathrm{LoRA}} = 2r_P$, and *Second* outputs. Rank 16 achieves the lowest mean BD-rate on each of the four metrics with 11.80M total LoRA parameters, compared with 17.69M and 29.49M for ranks 32 and 64, respectively. With the same hardware, datasets, and training recipe, larger ranks offer no further mean BD-rate gains, supporting $r_P = 16$ as our default.

A.4 User Study

Protocol and stimuli. We conducted an anonymous, reference-guided pairwise study with 30 consenting adults (18–34 years), yielding 1,920 formal judgments. Participants, mostly university students, were recruited online and volunteered without compensation. They used personal devices remotely in bright, quiet environments and knew their responses would be used for experimental analysis. All completed formal sessions were retained and practice data were excluded. We evaluated the eight methods listed in Table 11 on all 16 HEVC-B/C/D/E sequences (5/4/4/3), using 96-frame clips at native 24–60 fps (1.6–4.0 s). Gen2-VC-DCVC-UF, DCVC-UF and its SeedVR2 refinement shared the same bitrates. Other baselines used the nearest available bitrate at or above ours for each sequence.



(a) Source Preview (b) Choose Your Preference

Figure 10: User-study interface. (a) Source preview. (b) Anonymized candidate videos with playback controls and preference choices.

After four practice comparisons, participants completed 16 blocks of four comparisons (64 total), with breaks. Each block began with a complete source preview, available for replay throughout. Two anonymized candidates were then played synchronously twice before responses were enabled: left, right, or "no noticeable visual difference." Instructions emphasized fidelity to source content and structure, naturalness, artifacts, and temporal stability. Method identities and bitrates were hidden. Figure 10 shows the source preview and pairwise comparison interface.

Table 11: Preference among eight methods over 16 sequences at selected operating points.

Method	Pref. index ↑	95% CI	Mean BPP	BPP range
Gen2-VC-DCVC-UF	**90.00**	[87.56, 92.38]	0.00478	[0.00101, 0.00922]
DCVC-UF	65.17	[61.75, 68.74]	0.00478	[0.00101, 0.00922]
ECM-19.0	57.18	[53.54, 60.66]	0.00635	[0.00121, 0.01509]
SeedVR2 + DCVC-UF	51.69	[47.25, 56.28]	0.00478	[0.00101, 0.00922]
GVC1D	45.87	[42.62, 48.96]	0.00691	[0.00121, 0.01477]
GVC-RT	42.39	[39.34, 45.33]	0.00529	[0.00102, 0.00939]
GLC-Video	34.68	[31.15, 38.54]	0.00877	[0.00325, 0.01741]
YODA	13.01	[10.26, 15.89]	0.00850	[0.00328, 0.01976]

Pairing and randomization. For each sequence, the 28 ($\binom{8}{2} = 28$) method pairs were partitioned into seven units of four disjoint pairs. Participants received one unit per sequence, cyclically assigned by formal enrollment using frozen sequence-specific schedules. Sequence and within-block orders were randomized. Seeded left/right assignments placed each method eight times on each side per participant. Orientations reversed in successive seven-enrollment cycles. In the retained cohort, every method pair received 3–5 judgments per sequence.

Analysis. We fitted a Davidson model with method strengths $w_a > 0$, $w_b > 0$, a shared tie parameter $\nu > 0$, and a global left-position bias δ. For methods a (left) and b (right), the outcome probabilities were

$$(p_L, p_R, p_T) = \frac{\left(w_a e^{\delta/2},\ w_b e^{-\delta/2},\ \nu\sqrt{w_a w_b}\right)}{w_a e^{\delta/2} + w_b e^{-\delta/2} + \nu\sqrt{w_a w_b}}. \tag{5}$$

After maximum-likelihood fitting with a small L_2 penalty ($\lambda = 10^{-6}$), we set $\delta = 0$ in prediction and computed the position-neutral preference index

$$S_a = \frac{100}{7} \sum_{b \neq a} \left[P_0(a \succ b) + \tfrac{1}{2} P_0(a \sim b) \right]. \tag{6}$$

This model-based index summarizes relative preference, so 90.00 does not denote a 90% observed win rate. We obtained 95% percentile confidence intervals from 10,000 participant-cluster bootstrap refits, retaining each sampled participant's complete set of judgments.

Results. Gen2-VC-DCVC-UF achieved the highest preference index (90.00) among the eight tested methods. Its mean rate (0.00478 BPP) was joint lowest, matching DCVC-UF and SeedVR2 + DCVC-UF and remaining below the other five baselines.

A.5 Computational Complexity

Timing. We measure latency with synchronized PyTorch CUDA events after warmup. Baselines use the mean of ten calls, except GVC-RT, which uses the median. Latency t in milliseconds gives $1000n/t$ FPS, where n is the number of new frames per call in Table 12. Timing excludes model loading, disk I/O, and container serialization and does not capture all host-side overhead.

Profiled calls and bitstream coverage. Table 12 lists new frames per call and entropy-coding coverage. S^2VC and VoRTeC+ measurements cover P-frames and P-groups, respectively, excluding initial I-frame/group processing. Timings without entropy coding are for reference only. VoRTeC+ additionally excludes AIN color postprocessing.

Table 12: New frames per call and entropy-coding coverage in timing.

Method / stage	YODA GVC-RT DCVC-RT S^2VC	DCVC-UF	VoRTeC+	GLC-Video GVC1D	SeedVR2 VEnhancer-v2	DiQP Gen2-VC I-adapter	DiQP Gen2-VC P-adapter
New frames	1	8	8	1	9	1	4
Entropy coding	Included	Included	Excluded	Excluded	-	-	-

MACs, memory, and parameters. Table 13 summarizes the profiling tools and accounting rules. MACs are measured separately from latency and memory.

Table 13: Resource profiling libraries and measurement settings.

Resource	Implementation	Measurement settings
MACs	PyTorch `torch.utils.flop_counter.FlopCounterMode`	Codecs, SeedVR2, and Gen2-VC. Supported FLOPs divided by $2n$ give MACs per new frame. Codec counts use unfused reference execution.
MACs	Custom PyTorch forward hooks	DiQP and VEnhancer-v2. Count convolution, transposed-convolution, and linear modules, then divide by n.
Peak memory	PyTorch CUDA `reset_peak_memory_stats()`, `max_memory_allocated()`, `max_memory_reserved()`	Maximum allocated/reserved GPU memory over measured calls, including resident models. Gen2-VC uses two separate trials of 24 warmups and 30 measurements, reporting the maximum over 60 measured calls.
Parameters	PyTorch `numel()` with shared-tensor deduplication	Total model parameters and trainable adapter parameters. I/P totals share frozen parameters and should not be summed.

Hardware and scope. Table 14 reports component costs on one A100 80 GB GPU for 416×240, 832×480, 1280×720, and 1920×1080 inputs, with method-specific padding or resizing. Restoration methods and Gen2-VC report enhancement-only costs, with I/P profiled separately and P excluding I. SeedVR2 includes wavelet color correction as described in Appendix A.2.

Table 14: Component complexity on one A100 80 GB GPU. MACs: TMAC/frame. Memory: peak allocated/reserved (A/R), GiB. Parameters: total (trainable), M. S^2VC and VoRTeC+ use P-frame/group profiling. GLC-Video, GVC1D, and VoRTeC+ exclude entropy coding. Restoration methods and Gen2-VC report enhancement-only costs.

		FPS ↑		TMAC/frame ↓		Memory ↓	Params. ↓
Method	Res.	Enc.	Dec.	Enc.	Dec.	A/R	Total (Train.)
Distortion-oriented video codec							
DCVC-RT	240p	223.240	233.226	0.015	0.017	0.18/0.29	66.33
	480p	222.254	233.857	0.060	0.069	0.23/0.30	
	720p	212.467	173.954	0.138	0.160	0.31/0.42	
	1080p	130.670	119.125	0.312	0.362	0.49/0.74	
DCVC-UF	240p	2848.574	1998.211	0.013	0.015	0.37/0.40	162.72
	480p	1350.545	1014.879	0.054	0.058	0.54/0.62	
	720p	666.658	537.972	0.124	0.134	0.85/1.04	
	1080p	330.183	274.652	0.282	0.305	1.46/1.88	
VAE/tokenizer-based video codec (analysis/synthesis only, excluding entropy coding)							
GLC-Video	240p	53.393	55.358	0.139	0.223	0.75/1.29	289.33
	480p	32.391	21.277	0.519	0.830	1.22/2.01	
	720p	16.140	9.583	1.196	1.913	2.05/3.65	
	1080p	8.059	4.719	2.546	4.071	3.69/7.14	
GVC1D	240p	11.433	32.568	0.202	0.434	5.91/6.08	1800.93
	480p	10.663	18.904	0.856	1.833	8.38/8.44	
	720p	5.385	4.151	1.712	3.653	14.87/14.99	
	1080p	3.086	1.941	5.588	11.786	69.81/70.10	
VAE/tokenizer-based video codec (including entropy coding)							
GVC-RT	240p	224.368	106.927	0.017	0.039	0.32/0.43	135.17
	480p	208.311	114.131	0.064	0.145	0.36/0.49	
	720p	189.128	95.849	0.147	0.334	0.46/0.58	
	1080p	127.115	49.967	0.312	0.710	0.64/0.86	
Diffusion-based video codec							
YODA	240p	3.958	5.011	2.419	1.601	9.85/10.49	2055.13
	480p	2.860	3.920	4.469	2.951	11.64/12.93	
	720p	1.536	2.292	10.279	6.776	16.69/21.37	
	1080p	0.780	1.230	21.813	14.370	26.73/39.32	
S^2VC	240p	4.857	5.600	0.918	0.859	7.32/8.09	1842.29
	480p	2.148	2.483	3.532	3.310	8.25/9.41	
	720p	0.987	1.113	8.650	8.138	9.94/12.50	
	1080p	0.404	0.439	20.490	19.403	13.31/18.61	
VoRTeC+	240p	9.341	6.312	0.344	0.893	9.24/10.01	1802.72
	480p	0.977	0.482	1.288	3.551	14.62/17.54	
	720p	0.476	0.240	3.011	9.271	24.18/29.76	
	1080p	0.188	0.097	6.561	24.290	43.20/56.00	
Video restoration methods							
SeedVR2	240p	–	0.368	–	2.014	36.36/36.74	3642.12
	480p	–	0.344	–	8.028	39.94/40.55	
	720p	–	0.258	–	18.586	46.87/49.11	
	1080p	–	0.194	–	42.515	68.37/78.09	
DiQP	240p	–	3.050	–	0.589	1.48/2.15	79.37
	480p	–	1.695	–	1.178	1.48/2.15	
	720p	–	0.527	–	3.534	1.48/2.15	
	1080p	–	0.261	–	7.067	1.48/2.15	
VEnhancer-v2	240p	–	0.072	–	145.383	19.55/78.62	2496.59
	480p	–	0.072	–	145.383	19.58/78.62	
	720p	–	0.071	–	145.383	19.63/78.62	
	1080p	–	0.093	–	216.563	23.81/78.62	
Gen2-VC variants							
Gen2-VC I	240p	–	16.659	–	2.071	3.27/5.42	1551.79 (5.90)
	480p	–	5.234	–	8.205	3.29/12.56	
	720p	–	2.274	–	20.268	3.34/24.20	
	1080p	–	0.887	–	53.519	10.83/27.21	
Gen2-VC P	240p	–	12.807	–	2.430	5.21/7.37	1551.79 (5.90)
	480p	–	3.553	–	10.396	11.22/19.35	
	720p	–	1.477	–	27.122	21.67/39.08	
	1080p	–	0.557	–	77.546	45.25/69.17	

Complete Decoding Pipeline. Table 15 includes DCVC-UF decoding followed by I/P enhancement. On one GPU, the adapters run serially and share one resident Wan backbone. In the two-GPU pipeline, GPU A runs codec decoding and the I-adapter, while GPU B runs the P-adapter, with one resident Wan backbone per GPU. This split is enabled by keeping enhancement outside the codec's reference chain. In diffusion-based codecs that couple refinement with latent reference updates, temporal dependencies constrain such pipelining.

Table 15: Complete decoding costs of Gen2-VC-DCVC-UF. FPS is measured, memory is estimated, and MACs per frame are identical for both configurations.

	FPS ↑		TMAC/frame ↓	Estimated memory (GiB) ↓	
Res.	Single GPU	Two GPUs	Both	Single GPU	Two GPUs: A / B
240p	7.215	12.807	4.516	6.02	4.60 / 5.70
480p	2.112	3.553	18.660	12.43	6.69 / 12.12
720p	0.894	1.477	47.525	23.61	10.32 / 23.30
1080p	0.342	0.557	131.370	45.76	14.19 / 45.45

Cost interpretation. Counting shared weights once gives approximately 1.72B total parameters, below the totals of all three diffusion codecs in Table 14. Only 11.80M LoRA parameters are trained. Native DCVC-UF/RT decoding contributes little computation relative to I/P enhancement. Complete single-GPU throughput exceeds profiled VoRTeC+ decoding and all three restoration stages across resolutions, but trails S^2VC/YODA at 480p–1080p. Two GPUs improve throughput without reducing MACs or window buffering.

Buffering and output latency. Following Figure 3(b), frames retained for a *Second* estimate wait for the next overlapping call and four new inputs, adding buffering beyond native low-delay decoding. The reported FPS measures throughput, not per-frame output latency.

A.6 Per-Dataset BD-Rates

Tables 16, 17, and 18 report PSNR, MS-SSIM, LPIPS, and DISTS BD-rates for the 22 methods in Table 1. The default uses $r_I = r_P = 16$ and retains *Second* outputs. Gen2-VC-DCVC-UF combines strong LPIPS/DISTS performance with substantially better PSNR/MS-SSIM than the evaluated generative codecs across all six datasets. Training-free transfer also improves LPIPS and DISTS BD-rates over native DCVC-RT, ECM, VTM, and HM on all six datasets, demonstrating consistent perceptual gains across codec families.

Aggregation. Table 1 averages each metric's BD-rates equally across all six datasets using unrounded values. We report "–" for non-overlapping quality ranges or a six-dataset mean with any missing dataset result.

Table 16: BD-rate (%) ↓ relative to VTM-23.0 on UVG and MCL-JCV.

Method	UVG				MCL-JCV			
	PSNR	MS-SSIM	LPIPS	DISTS	PSNR	MS-SSIM	LPIPS	DISTS
Distortion-oriented video codec								
HM-18.0	39.78	46.17	30.67	13.05	44.54	53.35	40.10	16.98
VTM-23.0 (Anchor)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
ECM-19.0	−20.13	−19.09	−18.15	−14.64	**−22.37**	**−22.53**	−21.70	−16.56
DCVC-RT	−30.52	−21.19	−25.92	−6.23	−13.74	−8.64	−18.85	3.73
DCVC-UF	**−41.55**	**−37.28**	−51.85	−34.67	−20.98	−19.43	−46.03	−26.70
VAE/tokenizer-based video codec								
GLC-Video	858.40	570.84	−54.19	−93.04	504.62	291.25	−71.92	−92.94
GVC1D	413.50	322.89	−80.02	−94.49	219.05	159.68	**−89.27**	**−96.42**
GVC-RT	1127.30	175.55	−83.81	−94.27	846.23	132.34	−86.33	−92.37
Diffusion-based video codec								
YODA	1219.48	1139.05	−56.46	**−97.64**	712.53	634.29	−76.76	−95.76
S^2VC	384.01	167.87	−62.92	−83.40	257.01	122.46	−78.05	−88.40
VoRTeC+	547.74	300.17	−71.34	−91.12	412.10	209.04	−82.13	−92.41
Video restoration methods								
SeedVR2 + DCVC-RT	–	108.38	−19.29	−56.59	–	93.89	−19.25	−63.54
VEnhancer-v2 + DCVC-RT	–	917.69	4.18	−44.42	–	953.66	−3.31	−14.19
DiQP + DCVC-RT	63.86	19.51	49.66	63.10	102.89	57.52	61.19	61.04
SeedVR2 + DCVC-UF	–	42.65	−48.00	−68.08	–	40.23	−49.46	−76.18
VEnhancer-v2 + DCVC-UF	–	522.58	−23.22	−56.86	–	754.43	−36.55	−31.93
DiQP + DCVC-UF	11.94	−15.34	2.14	18.70	40.60	13.64	7.54	19.08
Gen2-VC variants								
Gen2-VC-HM	96.18	102.12	−24.58	−60.39	106.45	96.69	−2.55	−44.22
Gen2-VC-VTM	45.48	43.83	−70.56	−89.71	48.56	36.44	−64.82	−84.92
Gen2-VC-ECM	16.85	9.53	−82.89	−95.69	14.78	2.57	−80.83	−94.01
Gen2-VC-DCVC-RT	2.49	−0.29	−85.83	−96.49	14.28	5.22	−78.42	−92.86
Gen2-VC-DCVC-UF	−16.51	−24.65	**−89.33**	−97.24	1.09	−11.57	−83.97	−92.53

Table 17: BD-rate (%) ↓ relative to VTM-23.0 on HEVC-B and HEVC-C.

Method	HEVC-B				HEVC-C			
	PSNR	MS-SSIM	LPIPS	DISTS	PSNR	MS-SSIM	LPIPS	DISTS
Distortion-oriented video codec								
HM-18.0	36.90	45.90	26.20	11.81	36.35	45.26	25.71	11.04
VTM-23.0 (Anchor)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
ECM-19.0	−21.06	−20.53	−22.09	−21.36	−22.39	−20.34	−21.49	−12.70
DCVC-RT	−18.94	−19.05	−12.54	3.12	−14.72	−29.54	−16.72	10.38
DCVC-UF	**−28.37**	**−37.21**	−44.59	−36.60	**−30.75**	**−43.61**	−46.32	−31.59
VAE/tokenizer-based video codec								
GLC-Video	796.76	410.41	−62.19	−91.10	625.07	233.62	−56.91	−70.48
GVC1D	491.77	246.58	−83.82	−95.97	379.48	155.40	−76.34	−85.26
GVC-RT	713.26	124.20	−81.45	−92.69	293.91	66.85	−76.60	−87.28
Diffusion-based video codec								
YODA	1069.66	761.52	−65.77	**−98.17**	782.05	375.00	−55.19	−86.77
S^2VC	368.63	161.10	−69.98	−85.76	236.61	79.93	−60.74	−64.10
VoRTeC+	477.64	170.45	−81.51	−94.67	300.51	85.45	−79.00	−87.79
Video restoration methods								
SeedVR2 + DCVC-RT	–	624.27	−27.38	−66.17	–	499.08	−5.55	−28.96
VEnhancer-v2 + DCVC-RT	–	–	7.77	−40.98	–	492.70	1.34	−15.52
DiQP + DCVC-RT	63.22	12.07	29.82	41.13	59.80	14.62	31.05	78.71
SeedVR2 + DCVC-UF	–	12.83	−53.78	−74.96	–	399.81	−31.12	−45.95
VEnhancer-v2 + DCVC-UF	–	–	−34.37	−56.92	–	−4.19	−32.75	−39.88
DiQP + DCVC-UF	17.18	−19.93	−12.37	−2.19	8.41	−20.02	−15.07	24.16
Gen2-VC variants								
Gen2-VC-HM	95.87	81.20	−32.71	−51.25	98.72	70.36	−32.06	−49.65
Gen2-VC-VTM	46.34	24.10	−71.04	−86.31	70.41	28.38	−64.12	−79.73
Gen2-VC-ECM	25.32	1.68	−82.65	−94.91	42.61	6.43	−73.62	−86.92
Gen2-VC-DCVC-RT	5.95	−9.28	−83.80	−95.23	20.51	−12.71	−76.47	−85.98
Gen2-VC-DCVC-UF	−11.09	−30.02	**−88.83**	−96.66	−7.86	−32.91	**−84.54**	**−91.93**

Table 18: BD-rate (%) ↓ relative to VTM-23.0 on HEVC-D and HEVC-E.

Method	HEVC-D				HEVC-E			
	PSNR	MS-SSIM	LPIPS	DISTS	PSNR	MS-SSIM	LPIPS	DISTS
Distortion-oriented video codec								
HM-18.0	29.43	31.60	27.38	20.64	42.90	41.29	25.45	−4.23
VTM-23.0 (Anchor)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
ECM-19.0	−21.14	−14.77	−18.64	−14.97	−18.62	−17.01	−16.56	−11.43
DCVC-RT	−28.72	−40.29	−26.12	−12.40	−21.84	−17.40	−10.37	−5.25
DCVC-UF	**−48.44**	**−56.59**	−54.62	−45.96	**−52.88**	**−46.08**	−53.05	−42.83
VAE/tokenizer-based video codec								
GLC-Video	–	200.56	−46.46	−67.34	–	483.60	−45.93	−58.32
GVC1D	287.09	103.34	−74.94	−84.42	629.33	211.95	−87.09	−90.28
GVC-RT	160.30	29.58	−74.28	−85.09	1756.91	210.79	−89.46	−84.59
Diffusion-based video codec								
YODA	873.61	499.20	−23.20	−72.45	1248.55	812.81	−65.49	−93.05
S^2VC	160.73	39.98	−56.78	−63.01	439.50	186.83	−63.31	−80.82
VoRTeC+	170.18	39.12	−79.32	−86.49	547.79	285.36	−87.47	−92.27
Video restoration methods								
SeedVR2 + DCVC-RT	–	363.19	−24.59	57.76	–	476.80	−0.73	−41.45
VEnhancer-v2 + DCVC-RT	11.62	−18.04	−33.04	−30.38	–	351.98	52.50	−22.76
DiQP + DCVC-RT	12.15	−14.60	15.05	46.13	30.33	8.86	40.90	53.02
SeedVR2 + DCVC-UF	–	204.27	−49.01	−63.97	–	288.39	−35.68	−50.79
VEnhancer-v2 + DCVC-UF	−18.90	−42.93	−55.48	−52.69	–	367.31	1.85	−50.79
DiQP + DCVC-UF	−27.48	−43.05	−29.00	−7.45	−17.02	−27.74	−8.23	−0.03
Gen2-VC variants								
Gen2-VC-HM	82.34	50.06	−23.76	−42.04	84.43	73.17	−11.92	−40.83
Gen2-VC-VTM	36.86	8.90	−53.06	−70.03	56.97	35.91	−74.40	−83.88
Gen2-VC-ECM	40.33	9.59	−61.67	−79.04	31.92	12.71	−88.32	−94.45
Gen2-VC-DCVC-RT	−0.85	−28.53	−72.61	−83.95	9.95	1.89	−82.24	−96.05
Gen2-VC-DCVC-UF	−25.42	−46.54	**−81.20**	**−89.63**	−12.89	−22.51	**−92.06**	**−97.47**

A.7 Additional Visual Comparisons

A.7.1 Codec Comparisons

Figures 11–17 compare Gen2-VC-DCVC-UF with DCVC-UF, GLC-Video, GVC-RT, GVC1D and YODA, with selected examples also including S^2VC and VoRTeC+. Gen2-VC recovers finer detail while preserving source structure at the same bitrate as DCVC-UF and lower bitrates than the generative baselines shown. Red boxes mark the enlarged regions, BPP is reported above each method, and sampled frames are ordered from top to bottom.

A.7.2 Stages and Training-Free Transfer

Figures 18–22 compare native reconstructions with *I-only* and *I+P* using *First* or *Second* output selection, following Figure 3. The adapters transfer from DCVC-UF to ECM, VTM, HM, and DCVC-RT without retraining, recovering source details across learned and conventional codecs. Within each codec, all enhancement stages retain the native bitrate.

A.7.3 Video Restoration Comparisons

Figures 23 and 24 compare Gen2-VC with DiQP, SeedVR2, and VEnhancer-v2 on the same DCVC-UF reconstructions. These examples illustrate Gen2-VC's balance between detail recovery and source appearance under compression distortions. Red boxes identify the enlarged regions. Figure 24 additionally shows selected frames in temporal order within each group.



Figure 11: Low-bitrate codec comparison across sampled frames, example 1.

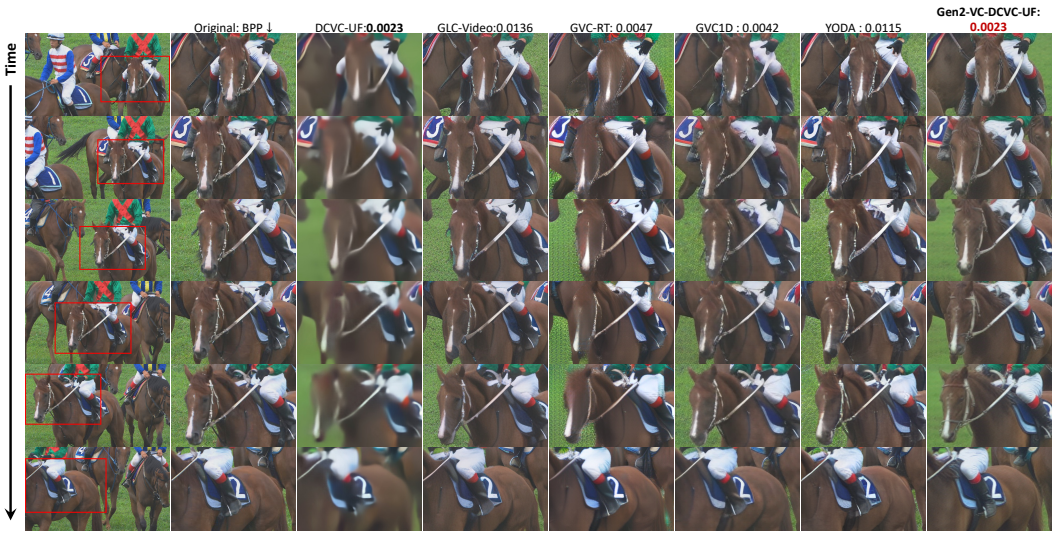


Figure 12: Low-bitrate codec comparison across sampled frames, example 2.

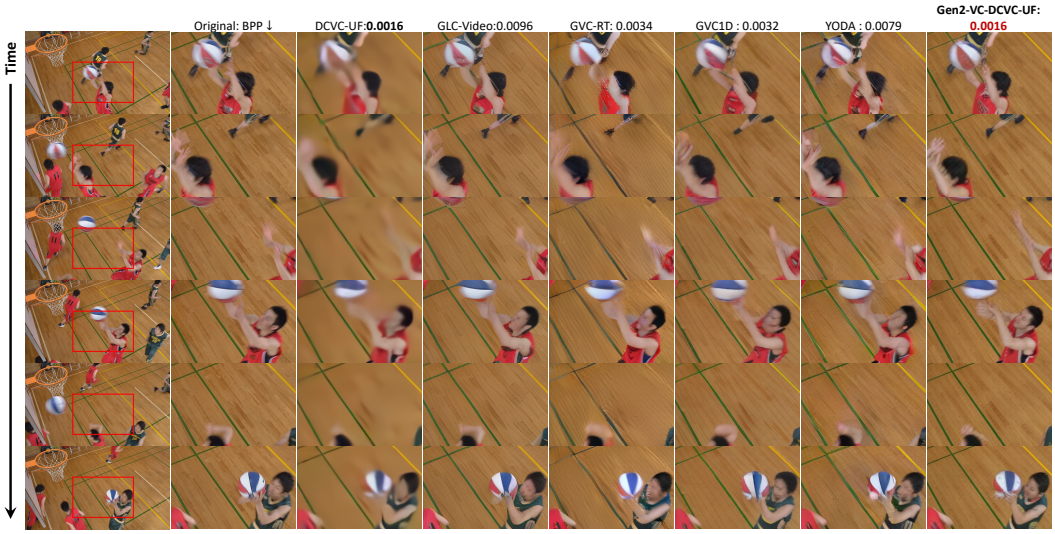


Figure 13: Low-bitrate codec comparison across sampled frames, example 3.

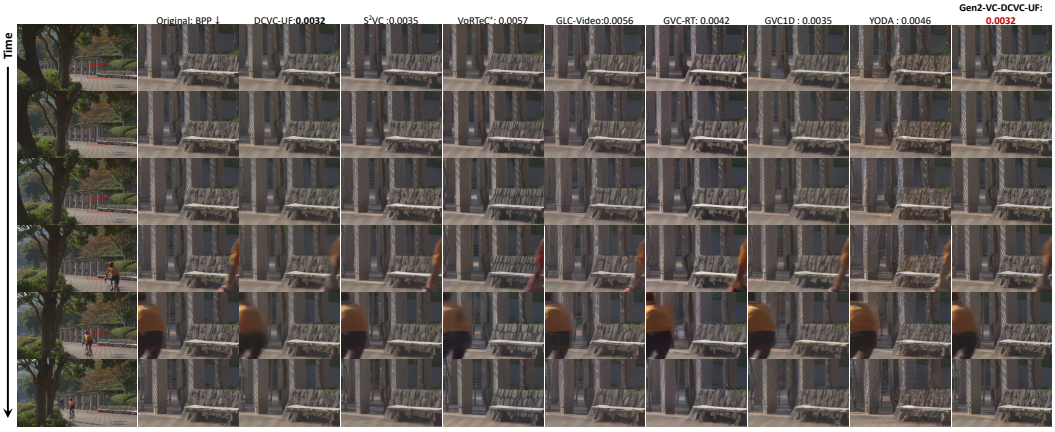


Figure 14: Low-bitrate codec comparison across sampled frames, example 4.



Figure 15: Low-bitrate codec comparison across sampled frames, example 5.

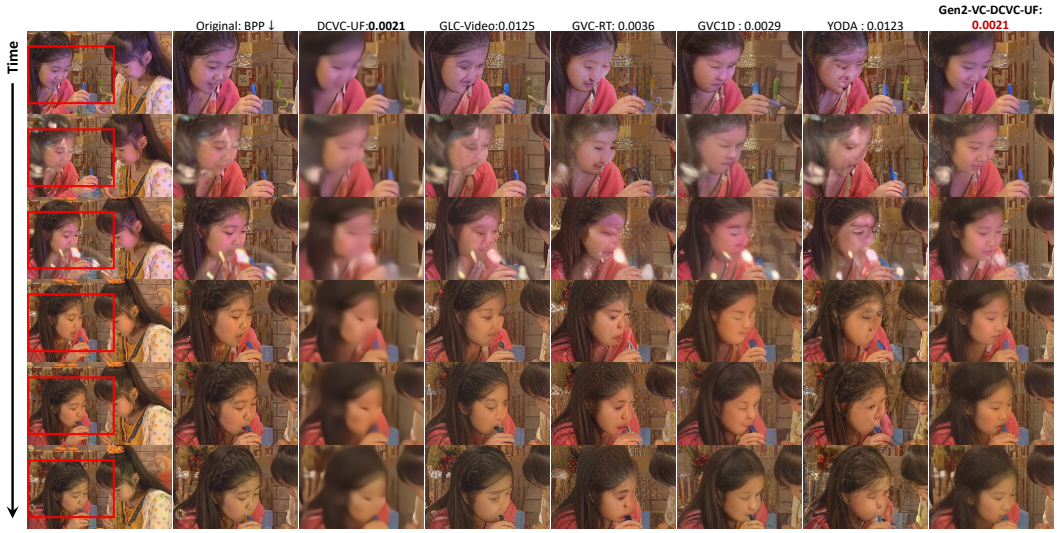


Figure 16: Low-bitrate codec comparison across sampled frames, example 6.

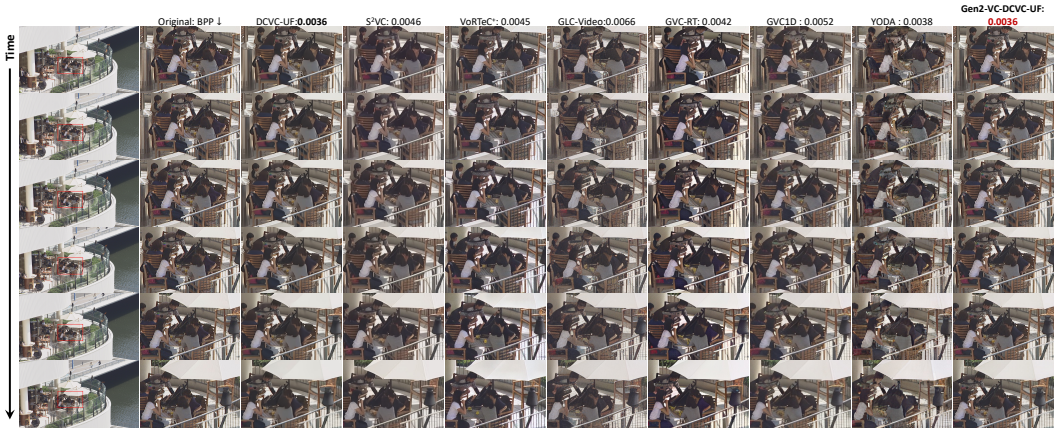


Figure 17: Low-bitrate codec comparison across sampled frames, example 7.

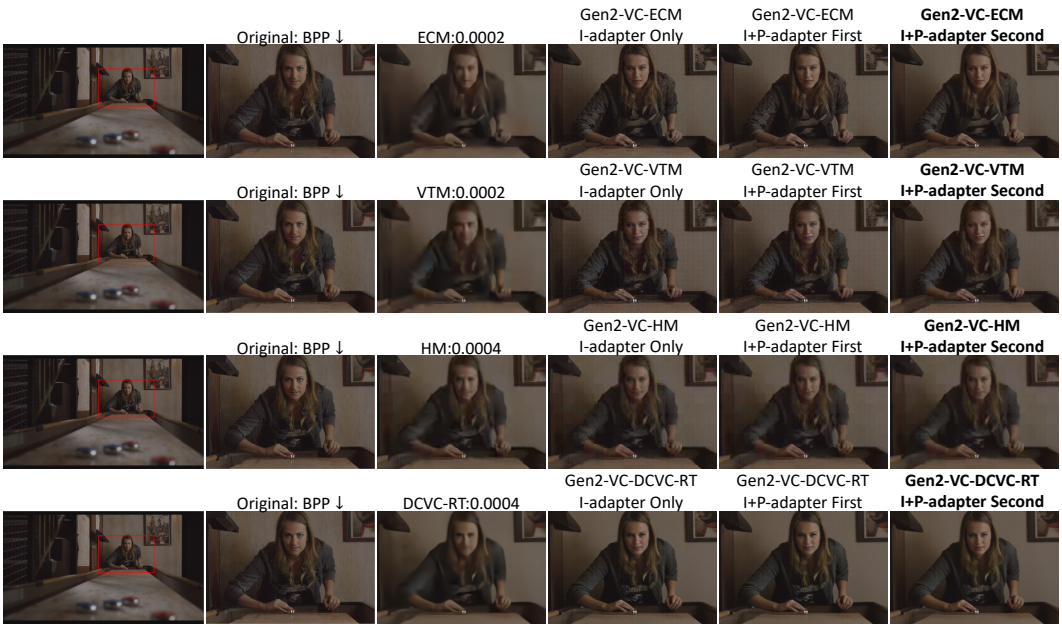


Figure 18: Stage comparisons after training-free transfer to four codecs, example 1.



Figure 19: Stage comparisons after training-free transfer to four codecs, example 2.



Figure 20: Stage comparisons after training-free transfer to four codecs, example 3.



Figure 21: Stage comparisons after training-free transfer to four codecs, example 4.



Figure 22: Stage comparisons across sampled frames for Gen2-VC-ECM and Gen2-VC-DCVC-RT.

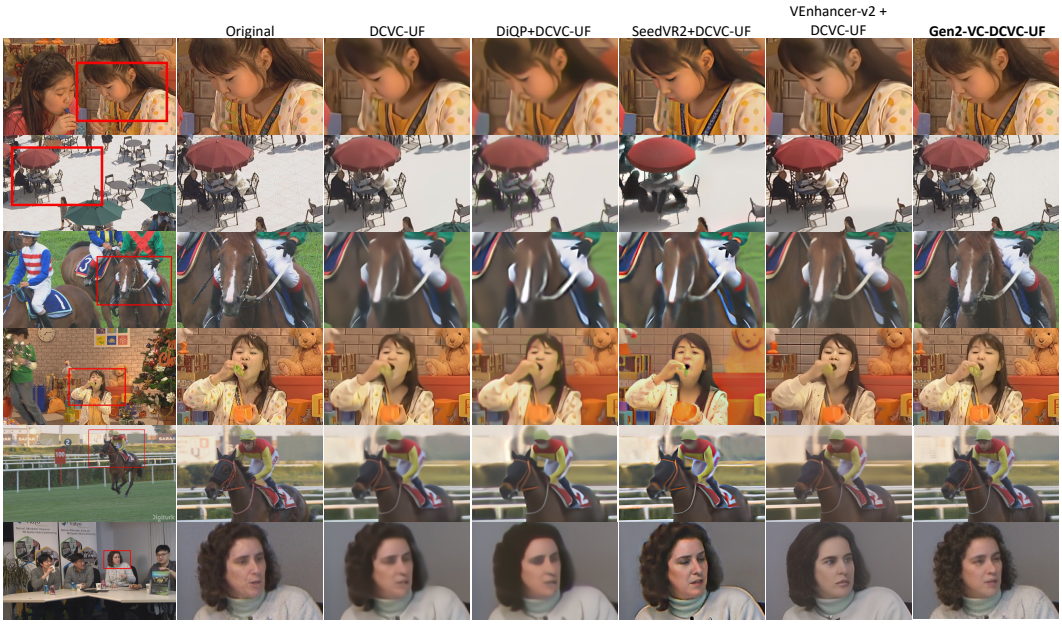


Figure 23: Restoration comparisons on DCVC-UF reconstructions for six selected frames.

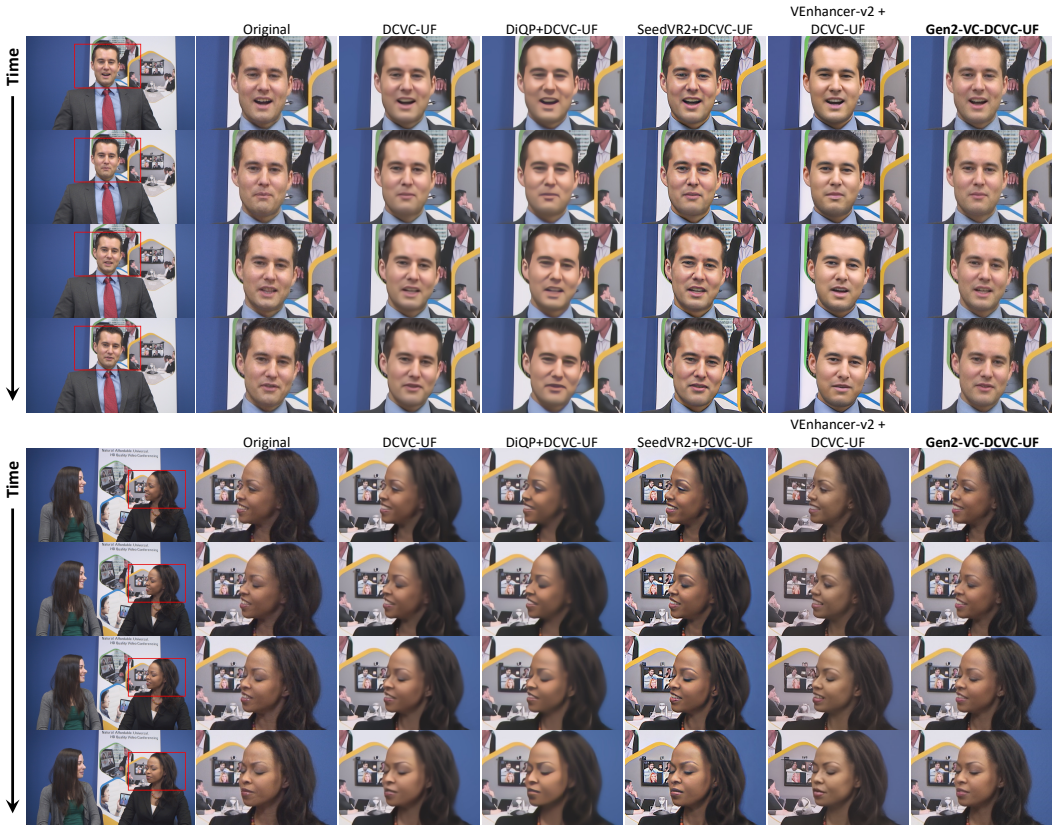


Figure 24: Restoration comparisons on DCVC-UF, with four sampled frames per sequence.